



Extended Web API





CONTENTS

Capabilities overview.....	8
About the LMS «Cargo».....	8
Overview.....	8
System access	8
Master data structure Element.....	9
Auxiliary data structure Row	10
Error information	11
Functions description.....	12
GetReferenceData: currencies	12
Request parameters:.....	12
Parameter description parameters	12
Answer:.....	12
Example	13
GetReferenceData: geographical object types	15
Request parameters:.....	15
Parameter description parameters	15
Answer:.....	15
GetReferenceData: cargo types	16
Request parameters:.....	16
Parameter description parameters	16
Answer:.....	16
Example	17
GetReferenceData: delivery types	18
Request parameters:.....	18
Parameter description parameters	18
Answer:.....	18
GetReferenceData: geography	19
Request parameters:.....	19
Parameter description parameters	19
Answer:.....	19
Example:	21
GetReferenceData: measurement units.....	23
Request parameters:.....	23
Parameter description parameters	23
Answer:.....	23
GetReferenceData: basic measurement units.....	24



Request parameters:.....	24
Parameter description parameters	24
Answer:.....	24
GetReferenceData: error codes.....	25
Request parameters:.....	25
Parameter description parameters	25
Answer:.....	25
GetReferenceData: client's contact persons.....	26
Request parameters:.....	26
Parameter description parameters	26
Answer:.....	26
GetReferenceData: cargoes description.....	28
Request parameters:.....	28
Parameter description parameters	28
Answer:.....	28
GetReferenceData: client's goods batches	29
Request parameters:.....	29
Parameter description parameters	29
Answer:.....	29
GetReferenceData: payers	30
Request parameters:.....	30
Parameter description parameters	30
Answer:.....	30
GetReferenceData: departments	31
Request parameters:.....	31
Parameter description parameters	31
Answer:.....	31
GetReferenceData: projects	32
Request parameters:.....	32
Parameter description parameters	32
Answer:.....	32
GetReferenceData: serial numbers of client goods.....	33
Request parameters:.....	33
Parameter description parameters	33
Answer:.....	33
GetReferenceData: cargo states tracking.....	34
Request parameters:.....	34
Parameter description parameters	34



Answer:.....	34
GetReferenceData: delivery methods	35
Request parameters:.....	35
Parameter description parameters	35
Answer:.....	35
GetReferenceData: payment methods.....	36
Request parameters:.....	36
Parameter description parameters	36
Answer:.....	36
GetReferenceData: urgencies	37
Request parameters:.....	37
Parameter description parameters	37
Answer:.....	37
GetReferenceData: VAT rates	38
Request parameters:.....	38
Parameter description parameters	38
Answer:.....	38
GetReferenceData: houses types	39
Request parameters:.....	39
Parameter description parameters	39
Answer:.....	39
GetReferenceData: contact information types	40
Request parameters:.....	40
Parameter description parameters	40
Answer:.....	40
GetReferenceData: rooms types	41
Request parameters:.....	41
Parameter description parameters	41
Answer:.....	41
GetReferenceData: house numbering types.....	42
Request parameters:.....	42
Parameter description parameters	42
Answer:.....	42
GetReferenceData: goods	43
Request parameters:.....	43
Parameter description parameters	43
Answer:.....	43
GetReferenceData: Goods, detail information.....	44



Request parameters:.....	44
Parameter description parameters	44
Answer:.....	44
Example:	46
GetReferenceData: Services	48
Request parameters:.....	48
Parameter description parameters	48
Answer:.....	48
GetReferenceData: warehouses	49
Request parameters:.....	49
Parameter description parameters	49
Answer:.....	49
GetReferenceData: characteristics of the client goods	50
Request parameters:.....	50
Parameter description parameters	50
Answer:.....	50
GetReferenceData: cargo state.....	51
Request parameters:.....	51
Parameter description parameters	51
Answer:.....	51
SaveDocuments: orders	51
Request parameters:.....	52
Parameter description parameters	52
Parameter description data.....	52
Answer:.....	56
Example:	57
Tracking: tracking of cargo state	61
Request parameters:.....	61
Parameter description parameters	61
Parameter description data.....	61
Answer:.....	61
Example:	64
PrintDocument: print document.....	66
Request parameters:.....	66
Answer:.....	66
Description of auxiliary type ResultPrint	66
Report: detailed bills.....	68
Request parameters:.....	68



Parameter description parameters	68
Answer:	68
Example:	71
SaveWaybillOffice: waybills with goods	74
Request parameters:	74
Type description order	74
Description of auxiliary type destinationinformation	75
Description of auxiliary type destinationaddress	75
Description of auxiliary type cargo.....	76
Description of auxiliary type documentsphoto	76
Description of auxiliary type products	77
Answer:	77
Description of auxiliary type resultstring	77
Example:	78
Calc: calculation of delivery cost.....	80
Request parameters:	80
Parameter description parameters	80
Parameter description data.....	80
Answer:	81
Example:	82
GetFormsForDocuments: waybill scan	86
Request parameters:	86
Parameter description parameters	86
Parameter description documents	86
Answer:	87
Example:	88
GetDocuments: document information.....	90
Request parameters:	90
Parameter description parameters	90
Parameter description data.....	90
Answer:	90
Example:	91
UpdateClientProducts: creating new or updating existing goods	96
Request parameters:	96
Parameter description parameters	96
Parameter description data.....	96
Answer:	97
Example:	98





CAPABILITIES OVERVIEW

This document is intended for developers and describes extended API capabilities to integrate with the LMS «Cargo».

ABOUT THE LMS «CARGO»

Logistics management system «Cargo» (the LMS «Cargo») is the product was developed by the company «PRO-Logistic». This system automates production activities of transport and logistics centers, logistic departments, courier and forwarding companies to optimize and increase operations profitability.

OVERVIEW

The extended web-service (API) of the LMS «Cargo» based on SOAP protocol for information exchange between different information systems. SOAP protocol uses the web services description language (WSDL) that describes model and structure in details to exchange data.

SYSTEM ACCESS

Functions list that is described in this document is available in the publication:

<http://web.cse.ru/1c/ws/Web1C.1cws?wsdl>

Use the following credential to read the publication: *login*: web, *password*: web.

You must know the user name and password to call system functions. If you have already a personal account, you must use the same credential to call the web-service. Otherwise, please contact your manager of client service to get them.



MASTER DATA STRUCTURE ELEMENT

Two parameters types are used to pass and get data: simple string (string) that are used, for example, to transfer credential (user name and password) and universal data structures «Element» and «Row» that allows to pass and get data of any complexity using nested structure of «Element» and «Row». Everywhere if simple type isn't used, it means, structure «Element» is used.

The structure «Element» consists of the following properties:

Property	Type	Required	Description
Key	String	Yes	«Key» of element. It usually denotes the name of parameter/field/table data or a number/code of data object (for example, the document number)
Value	Any simple		Passed value
CultureSpecificValues	Element - list		Uses to return the response in different languages. The property «Key» of returned structures contains a language, and property «Value» – the value in this language.
ValueType	String		Describes what the value type is passed/get in properties «Value»/«Values» and «CultureSpecificValues». It can be one of the following values: «string», «boolean», «float» or «dateTime»
Properties	Element - list		List of passed properties (usually settings). Set the property «Key» for every property in every structure «Element» of this list
Fields	Element - list		List of passed fields (usually references or documents data). Set property «Key» for every field name in every structure «Element» of this list.
List	Element - list		Data list of the one type (usually references, documents or table rows)
Tables	Element - list		List of passed tables (usually documents tables). Set property «Key» for every table name in every structure of this list.
Values	Any simple - list		Passed values. Use this property instead of «Value» to pass a list and at the same time not to create structure «Element» in property «List».
Rows	Row - list		List of passed data as the structure «Row» (it uses for simplified tabular data transfer)
BData	hexBinary		Passed binary data, for example, file content

Any-volume information transfer can be for element – lists: parameters, data, lists and tables.



AUXILIARY DATA STRUCTURE ROW

The property «Row» is able to use instead of property «Table» of structure «Element» when it is required to transfer a large-volume information as a tabular. It allows not to create a set of nested structures «Element» and to reduce the volume of passed data significantly. Structure «Row» includes the passed data list without name and type of every of them.

The structure «Row» consists of the following properties:

Property	Type	Required	Description
Cells	Any simple - list		Passed values. Every value can have its own type. Only developer controls the passed data composition and types.
Rows	Row - list		List of passed data as the structure «Row» (it uses to transfer nested tabular data)
Properties	Element - list		List of passed properties (usually settings). Set the property «Key» for every property name in every structure «Element» of this list.



ERROR INFORMATION

In case of an error, for any reason, the structure «Element» is also used to transfer error information to a client. In some cases the error can remain not only in the response of the root structure «Element», but also in the nested structure's response. For example, to create the request for saving of several documents at once. In response, request returns the structure where the root element will contain the general error information but the structure «Element» - errors associated with every documents from the request.

All errors are stored in property «Properties» of the structure «Element» and have «Key» is equal «Error»; errors codes (may be several) are stored in the property «List» as the structure «Element», where «Key» is equal «Description» and «Value» contains the error code. Besides, additional information can be registered for some errors, for example, which code or number caused the problem. In this case this information is stored in the list «List» of the structure «Description» where every element has «Key» is equal «Info» and «Value» contains additional information. How to get error description by the code and error list read more in function description [GetReferenceData: errors codes](#). Look at the list of all possible errors in the file «Errors of the web service and HTTP API access to the LMS «Cargo».pdf».



FUNCTIONS DESCRIPTION

GETREFERENCEDATA: CURRENCIES

The function allows to get a list of currencies that is used in other functions called by the web service.

REQUEST PARAMETERS:

string **login** user name
string **password** password
Element **parameters** parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter(Key)	Value type	Required	Description
Reference	string	Yes	Reference name, it must be equal «Currencies»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «Currencies» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Global Unique Identifier (GUID) of the reference element
Value	string	Yes	Currency short name
Fields	Element - list		Other properties of the reference element

Properties values passed through «**Fields**»:

Parameter(Key)	Value type	Required	Description
FullName	string		Currency full name
Default	boolean		True if this currency is used by default



EXAMPLE

Request:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body> <m:GetReferenceData xmlns:m="http://www.cargo3.ru">
    <m:login xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">LOGIN</m:login>
    <m:password xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">PASSWORD</m:password>
    <m:parameters xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <m:Key>Parameters</m:Key>
      <m:List>
        <m:Key>Reference</m:Key>
        <m:Value xsi:type="xs:string">Currencies</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:List>
    </m:parameters>
  </m:GetReferenceData></soap:Body>
</soap:Envelope>
```

Answer:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body> <m:GetReferenceDataResponse xmlns:m="http://www.cargo3.ru">
    <m:return xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <m:Key>Currencies</m:Key>
      <m:List>
        <m:Key>e6853796-4421-11dc-9497-0015170f8c09</m:Key>
        <m:Value xsi:type="xs:string">EUR</m:Value>
        <m:ValueType>string</m:ValueType>
        <m:Fields>
          <m:Key>FullName</m:Key>
          <m:Value xsi:type="xs:string">Евро</m:Value>
          <m:ValueType>string</m:ValueType>
        </m:Fields>
        <m:Fields>
          <m:Key>IsFolder</m:Key>
          <m:Value xsi:type="xs:boolean">>false</m:Value>
          <m:ValueType>boolean</m:ValueType>
        </m:Fields>
        <m:Fields>
          <m:Key>Parent</m:Key>
          <m:Value xsi:type="xs:string"/>
          <m:ValueType>string</m:ValueType>
        </m:Fields>
        <m:Fields>
          <m:Key>Level</m:Key>
          <m:Value xsi:type="xs:decimal">0</m:Value>
          <m:ValueType>float</m:ValueType>
        </m:Fields>
        <m:Fields>
          <m:Key>Default</m:Key>
          <m:Value xsi:type="xs:boolean">>false</m:Value>
          <m:ValueType>boolean</m:ValueType>
        </m:Fields>
      </m:List>
      <m:List>
        <m:Key>ff3f7c38-4430-11dc-9497-0015170f8c09</m:Key>
        <m:Value xsi:type="xs:string">RUR</m:Value>
        <m:ValueType>string</m:ValueType>
        <m:Fields>
          <m:Key>FullName</m:Key>
          <m:Value xsi:type="xs:string">Рубль РФ</m:Value>
          <m:ValueType>string</m:ValueType>
        </m:Fields>
        <m:Fields>
          <m:Key>IsFolder</m:Key>
          <m:Value xsi:type="xs:boolean">>false</m:Value>
          <m:ValueType>boolean</m:ValueType>
        </m:Fields>
      </m:List>
    </m:return>
  </m:GetReferenceDataResponse>
</soap:Body>
</soap:Envelope>
```



```

        <m:Key>Parent</m:Key>
        <m:Value xsi:type="xs:string"/>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Level</m:Key>
        <m:Value xsi:type="xs:decimal">0</m:Value>
        <m:ValueType>float</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Default</m:Key>
        <m:Value xsi:type="xs:boolean">true</m:Value>
        <m:ValueType>boolean</m:ValueType>
    </m:Fields>
</m:List>
<m:List>
    <m:Key>e6853795-4421-11dc-9497-0015170f8c09</m:Key>
    <m:Value xsi:type="xs:string">USD</m:Value>
    <m:ValueType>string</m:ValueType>
    <m:Fields>
        <m:Key>FullName</m:Key>
        <m:Value xsi:type="xs:string">Доллар США</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>IsFolder</m:Key>
        <m:Value xsi:type="xs:boolean">false</m:Value>
        <m:ValueType>boolean</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Parent</m:Key>
        <m:Value xsi:type="xs:string"/>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Level</m:Key>
        <m:Value xsi:type="xs:decimal">0</m:Value>
        <m:ValueType>float</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Default</m:Key>
        <m:Value xsi:type="xs:boolean">false</m:Value>
        <m:ValueType>boolean</m:ValueType>
    </m:Fields>
</m:List>
</m:return>
</m:GetReferenceDataResponse></soap:Body>
</soap:Envelope>

```



GETREFERENCEDATA: GEOGRAPHICAL OBJECT TYPES

The function allows to get geographical object types (country, region, city and etc.) that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «TypesOfGeographicObjects»

ANSWER:

[Element](#)

«Element» has two properties: the property «**Key**» contains the value «TypesOfGeographicObjects» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Id of the geographical object type
Value	string	Yes	Name of the geographical object type
Fields	Element - list		Other properties of the reference element

Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
Country	boolean		True if this geographical object type is a country
Town	boolean		True if this geographical object type is a human settlement



GETREFERENCEDATA: CARGO TYPES

The function allows to get a list of cargo types (cargo, documents and etc.) that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «TypesOfCargo»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «TypesOfCargo» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Global Unique Identifier (GUID) of the reference element
Value	string	Yes	Cargo type name
Fields	Element - list		Other properties of the reference element

Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
Default	boolean		True if this cargo type is used by default



EXAMPLE

Request

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body> <m:GetReferenceData xmlns:m="http://www.cargo3.ru">
    <m:login xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">LOGIN</m:login>
    <m:password xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">PASSWORD</m:password>
    <m:parameters xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <m:Key>Parameters</m:Key>
      <m:List>
        <m:Key>Reference</m:Key>
        <m:Value xsi:type="xs:string">TypesOfCargo</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:List>
    </m:parameters>
  </m:GetReferenceData></soap:Body>
</soap:Envelope>
```

Answer:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body> <m:GetReferenceDataResponse xmlns:m="http://www.cargo3.ru">
    <m:return xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <m:Key>TypesOfCargo</m:Key>
      <m:List>
        <m:Key>81dd8a13-8235-494f-84fd-9c04c51d50ec</m:Key>
        <m:Value xsi:type="xs:string">Документы</m:Value>
        <m:ValueType>string</m:ValueType>
        <m:Fields>
          <m:Key>Default</m:Key>
          <m:Value xsi:type="xs:boolean">true</m:Value>
          <m:ValueType>boolean</m:ValueType>
        </m:Fields>
      </m:List>
      <m:List>
        <m:Key>4aab1fc6-fc2b-473a-8728-58bcd4ff79ba</m:Key>
        <m:Value xsi:type="xs:string">Груз</m:Value>
        <m:ValueType>string</m:ValueType>
        <m:Fields>
          <m:Key>Default</m:Key>
          <m:Value xsi:type="xs:boolean">false</m:Value>
          <m:ValueType>boolean</m:ValueType>
        </m:Fields>
      </m:List>
    </m:return>
  </m:GetReferenceDataResponse></soap:Body>
</soap:Envelope>
```



GETREFERENCEDATA: DELIVERY TYPES

The function allows to get delivery types that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «Urgencies»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «deliverytype» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Element index from 1S
Value	string	Yes	Name
Fields	Element - list		Other properties of the reference element

Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
Information	string		Extra information
Default	boolean		True if this delivery type is used by default



GETREFERENCEDATA: GEOGRAPHY

The function allows to get a list of geographic objects that are used in other functions called by the web service.

The geography reference is hierarchic. That means, all geography data are stored as a tree that at the bottom has countries list but on the top – cities and settlements. The function can return data for only one branch and only one level. Alternatively, function can return the result by geographical object name.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «Geography»
Search	string		Geographic object name for search. Name must be the same as the object name in the LMS «Cargo». When the function searches by name it cannot search in a group (parameter « InGroup » is ignored)
InGroup	string		Group name where the function selects geographic objects. If it gets unknown parameter or an empty string, the function will select data in the geography's root among countries.

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «Geography» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Global Unique Identifier (GUID) of the reference element
Value	string	Yes	Geographical object name
Fields	Element - list		Other properties of the reference element



Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
Default	boolean		True if this element is used by default
Parent	string		GUID (Globally Unique Identifier) of this element's parent. If the element doesn't have parent then the function returns an empty string.
Level	float	Yes	Nesting level of this element in the geography reference
Settlement	boolean		True if this object is a human settlement
State	boolean		True if this object is a country
Type	float	Yes	Geographical object type (see the chapter GetReferenceData: geographical object types)
IATA	string		Code IATA
ICAO	string		Code ICAO
PrimaryState	boolean		True if this object is a country and is used by default
HaveDependentObjects	boolean		True if this object has a subordinate geography
HaveDependentSubway Stations	boolean		True if this object has an information about metro station



EXAMPLE:

Request (looking for the city Moscow):

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body> <m:GetReferenceData xmlns:m="http://www.cargo3.ru">
    <m:login xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">LOGIN</m:login>
    <m:password xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">PASSWORD</m:password>
    <m:parameters xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <m:Key>Parameters</m:Key>
      <m>List>
        <m:Key>Reference</m:Key>
        <m:Value xsi:type="xs:string">Geography</m:Value>
        <m:ValueType>string</m:ValueType>
      </m>List>
      <m>List>
        <m:Key>Search</m:Key>
        <m:Value xsi:type="xs:string">Москва r</m:Value>
        <m:ValueType>string</m:ValueType>
      </m>List>
    </m:parameters>
  </m:GetReferenceData></soap:Body>
</soap:Envelope>
```

Answer:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body> <m:GetReferenceDataResponse xmlns:m="http://www.cargo3.ru">
    <m:return xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <m:Key>Geography</m:Key>
      <m>List>
        <m:Key>cf862f56-442d-11dc-9497-0015170f8c09</m:Key>
        <m:Value xsi:type="xs:string">Москва r</m:Value>
        <m:ValueType>string</m:ValueType>
        <m:Fields>
          <m:Key>Default</m:Key>
          <m:Value xsi:type="xs:boolean">true</m:Value>
          <m:ValueType>boolean</m:ValueType>
        </m:Fields>
        <m:Fields>
          <m:Key>Parent</m:Key>
          <m:Value xsi:type="xs:string">f2cd6487-4422-11dc-9497-0015170f8c09</m:Value>
          <m:ValueType>string</m:ValueType>
        </m:Fields>
        <m:Fields>
          <m:Key>IsFolder</m:Key>
          <m:Value xsi:type="xs:boolean">>false</m:Value>
          <m:ValueType>boolean</m:ValueType>
        </m:Fields>
        <m:Fields>
          <m:Key>Level</m:Key>
          <m:Value xsi:type="xs:decimal">1</m:Value>
          <m:ValueType>float</m:ValueType>
        </m:Fields>
        <m:Fields>
          <m:Key>Settlement</m:Key>
          <m:Value xsi:type="xs:boolean">true</m:Value>
          <m:ValueType>boolean</m:ValueType>
        </m:Fields>
        <m:Fields>
          <m:Key>State</m:Key>
          <m:Value xsi:type="xs:boolean">>false</m:Value>
          <m:ValueType>boolean</m:ValueType>
        </m:Fields>
        <m:Fields>
          <m:Key>Type</m:Key>
          <m:Value xsi:type="xs:string">3</m:Value>
          <m:ValueType>string</m:ValueType>
        </m:Fields>
      </m:List>
    </m:return>
  </m:GetReferenceDataResponse>
</soap:Body>
</soap:Envelope>
```



```

        <m:Key>KLADR</m:Key>
        <m:Value xsi:type="xs:string">7 700 000 000 000 000 000 000 000</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>ID</m:Key>
        <m:Value xsi:type="xs:string">cf862f56-442d-11dc-9497-0015170f8c09</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>AltCode</m:Key>
        <m:Value xsi:type="xs:string">MOW</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>PrimaryState</m:Key>
        <m:Value xsi:type="xs:boolean">>false</m:Value>
        <m:ValueType>boolean</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>HaveDependentObjects</m:Key>
        <m:Value xsi:type="xs:boolean">>true</m:Value>
        <m:ValueType>boolean</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>HaveDependentSubwayStations</m:Key>
        <m:Value xsi:type="xs:boolean">>true</m:Value>
        <m:ValueType>boolean</m:ValueType>
    </m:Fields>
</m:List>
</m:return>
</m:GetReferenceDataResponse></soap:Body>
</soap:Envelope>

```



GETREFERENCEDATA: MEASUREMENT UNITS

The function allows to get a list of measurement units that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «UnitsOfMeasurement»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «UnitsOfMeasurement» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	GUID (Globally Unique Identifier) of the measurement unit
Value	string	Yes	Name of the measurement unit



GETREFERENCEDATA: BASIC MEASUREMENT UNITS

The function allows to get a list of measurement units that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «BaseUnitsOfMeasurement»

ANSWER:

Element

«Element» has two properties: property «**Key**» contains the value «BaseUnitsOfMeasurement» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	GUID (Globally Unique Identifier) of the measurement unit
Value	string	Yes	Name of the measurement unit



GETREFERENCEDATA: ERROR CODES

The function allows to get a list of error codes that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «ErrorCodes»
Search	string		Error code. The function returns information on the requested code.

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «ErrorCodes» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Error code
Value	string	Yes	Error description



GETREFERENCEDATA: CLIENT'S CONTACT PERSONS

The function allows to get a list of client's contact persons that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «Contacts»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «Contacts» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Global Unique Identifier (GUID) of the reference element
Value	string	Yes	Contact full name
Fields	Element - list		Other properties of the reference element
List	Element - list		Contact information of the contact person

Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
FirstName	string		First name
LastName	string		Surname
Patronymic	string		Patronymic
Department	string		Department
Project	string		Assigned project
DeletionMark	boolean		True if this contact is marked on deleting in the LMS «Cargo»



Every element with contact information (from the list «List») has two properties: the property «Key» contains description of the contact information type and the property «Value» contains the contact information value. Also every element in list «List» has the following properties:

Parameter (Key)	Value type	Required	Description
Type	string		Type code of the contact information (see the chapter GetReferenceData: types of contact information)
SubType	string		Subtype code of the contact information (see the chapter GetReferenceData: types of contact information)
Geography	string		GUID (Globally Unique Identifier) of the geographical object (see the chapter GetReferenceData: geography)



GETREFERENCEDATA: CARGOES DESCRIPTION

The function allows to get a list of cargoes descriptions that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «CargoDescriptions»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «CargoDescriptions» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Global Unique Identifier (GUID) of the reference element
Value	string	Yes	Cargo description
Fields	Element - list		Other properties of the reference element

Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
DeletionMark	boolean		True if this description is marked on deleting in the LMS «Cargo» (as unused or outdated)



GETREFERENCEDATA: CLIENT'S GOODS BATCHES

The function allows to get a list of client's goods batches that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «PartiesOfItems»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «PartiesOfItems» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Global Unique Identifier (GUID) of the reference element
Value	string	Yes	Name of the goods batch
Fields	Element - list		Other properties of the reference element

Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
ID	string		Batch ID
Date	dateTime		Arrival date of the batch
ExpirationDate	dateTime		Best before date of the batch



GETREFERENCEDATA: PAYERS

The function allows to get possible payers (sender, recipient, third party) that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «Payers»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «Payers» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Payer ID
Value	string	Yes	Payer name
Fields	Element - list		Other properties of the reference element

Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
Default	boolean		True if this payer is used by default



GETREFERENCEDATA: DEPARTMENTS

The function allows to get a list of client's departments that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «Departments»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «Departments» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Global Unique Identifier (GUID) of the reference element
Value	string	Yes	Department name
Fields	Element - list		Other properties of the reference element

Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
DeletionMark	boolean		True if this department is marked on deleting in the LMS «Cargo» (as unused or outdated)
Default	boolean		True if this department is used by default



GETREFERENCEDATA: PROJECTS

The function allows to get a list of client's projects that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «Projects»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «Projects» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Global Unique Identifier (GUID) of the reference element
Value	string	Yes	Project name
Fields	Element - list		Other properties of the reference element

Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
DeletionMark	boolean		True if this project is marked on deleting in the LMS «Cargo» (as unused or outdated)
Default	boolean		True if this project is used by default



GETREFERENCEDATA: SERIAL NUMBERS OF CLIENT GOODS

The function allows to get serial numbers' list of client goods that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «SerialNumbersOfItems»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «SerialNumbersOfItems» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Global Unique Identifier (GUID) of the reference element
Value	string	Yes	Serial number
Fields	Element - list		Other properties of the reference element

Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
ID	string		ID
ExpirationDate	dateTime		Best before date



GETREFERENCEDATA: CARGO STATES TRACKING

The function allows to get a list of all available states returned by function *Tracking* .

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «CargoStates»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «CargoStates» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Global Unique Identifier (GUID) of the reference element
Value	string	Yes	State name



GETREFERENCEDATA: DELIVERY METHODS

The function allows to get a list of delivery methods that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «ShippingMethods»

ANSWER:

Element

«Element» has two properties: the property «Key» contains the value «ShippingMethods» and the property «List» contains reference elements. The property «Properties» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «List» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Global Unique Identifier (GUID) of the reference element
Value	string	Yes	Name of the delivery method
Fields	Element - list		Other properties of the reference element

Properties values passed through «Fields»:

Parameter (Key)	Value type	Required	Description
DimensionalWeightFactor	float		The volume weight coefficient that is used for this delivery method
Default	boolean		True if this delivery method is used by default



GETREFERENCEDATA: PAYMENT METHODS

The function allows to get a list of available payment methods (cash payment, cashless payment, etc.) that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «PaymentMethods»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «PaymentMethods» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Id of the payment method
Value	string	Yes	Name of the payment method
Fields	Element - list		Other properties of the reference element

Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
Default	boolean		True if this payment method is used by default



GETREFERENCEData: URGENCIES

The function allows to get urgency types that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «Urgencies»

ANSWER:

Element

«Element» has two properties: the property «Key» contains the value «Urgencies» and the property «List» contains reference elements. The property «Properties» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «List» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Global Unique Identifier (GUID) of the reference element
Value	string	Yes	Urgency name
Fields	Element - list		Other properties of the reference element

Properties values passed through «Fields»:

Parameter (Key)	Value type	Required	Description
Information	string		Extra information
Default	boolean		True if this urgency is used by default



GETREFERENCEDATA: VAT RATES

The function allows to get a list of VAT rates that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «VATRates»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «VATRates» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Id of VAT rate
Value	string	Yes	Description of VAT rate



GETREFERENCEDATA: HOUSES TYPES

The function allows to get available houses types (house, building, etc) that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «TypesOfHouses»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «TypesOfHouses» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Id of the house type
Value	string	Yes	Name of the house type
Fields	Element - list		Other properties of the reference element

Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
Default	boolean		True if this house type is used by default



GETREFERENCEDATA: CONTACT INFORMATION TYPES

The function allows to get a list of available contact information types (address, phone number, website, etc) that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «TypesOfContactInformation». Use the name «SubTypesOfContactInformation» to get subtypes of contact information reference.

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «TypesOfContactInformation» («SubTypesOfContactInformation» for subtypes) and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Id of the contact information type
Value	string	Yes	Name of the contact information type



GETREFERENCEDATA: ROOMS TYPES

The function allows to get available rooms types (flat, office) that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «TypesOfPremises»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «TypesOfPremisesTypesOfPremises» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Id of the room type
Value	string	Yes	Name of the room type
Fields	Element - list		Other properties of the reference element

Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
Default	boolean		True if this room type is used by default



GETREFERENCEDATA: HOUSE NUMBERING TYPES

The function allows to get available house numbering types (block, building, etc) that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «TypesOfBuildings»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «TypesOfBuildings» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Id of the house numbering type
Value	string	Yes	Name of the house numbering type
Fields	Element - list		Other properties of the reference element

Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
Default	boolean		True if this house numbering type is used by default



GETREFERENCEDATA: GOODS

The function allows to get a list of client goods (Internet shop) that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «Items»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «Items» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Global Unique Identifier (GUID) of the reference element
Value	string	Yes	Goods name
Fields	Element - list		Other properties of the reference element

Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
FullName	string		Full name
Article	string		Goods article
ID	string		Goods ID
Measure	string		GUID (Globally Unique Identifier) of the goods measurement unit



GETREFERENCEData: GOODS, DETAIL INFORMATION

The function allows to get a list of client goods (Internet shop) that are used in other functions called by the web service. Every item contains detail information about goods.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «ItemsInStock»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «ItemsInStock» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Global Unique Identifier (GUID) of the reference element
Value	string	Yes	Goods name
Fields	Element - list		Other properties of the reference element
Properties	Element - list		Additional goods characteristics

Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
FullName	string		Goods full name
Article	string		Goods article
ID	string		Goods ID
Balance	float		Goods total in a warehouse
Reserve	float		Reserved goods quantity
Measure	string		GUID (Globally Unique Identifier) of the goods measurement unit

Additional characteristics of goods passed through property «**Properties**»:

Parameter (Key)	Value type	Required	Description
Characteristic	string		Element GUID (Globally Unique Identifier) of the reference «Characteristics»



Party string Element GUID (Globally Unique Identifier) of the reference «Goods batches»

SerialNumber string Element GUID (Globally Unique Identifier) of the reference «Serial numbers»

Every characteristic (characteristic, batch number, serial number) has extra information in the property «**Fields**»:

Parameter (Key)	Value type	Required	Description
Name	string		Characteristic/goods batch/serial number name
FullName	string		Full name (only for a characteristic)



EXAMPLE:

Request:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body> <m:GetReferenceData xmlns:m="http://www.cargo3.ru">
    <m:login xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">LOGIN</m:login>
    <m:password xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">PASSWORD</m:password>
    <m:parameters xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <m:Key>Parameters</m:Key>
      <m:List>
        <m:Key>Reference</m:Key>
        <m:Value xsi:type="xs:string">ItemsInStock</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:List>
    </m:parameters>
  </m:GetReferenceData></soap:Body>
</soap:Envelope>
```

Answer:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body> <m:GetReferenceDataResponse xmlns:m="http://www.cargo3.ru">
    <m:return xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <m:Key>ItemsInStock</m:Key>
      <m:List>
        <m:Key>3e56f417-40b1-11e3-801a-001e670c42dc</m:Key>
        <m:Value xsi:type="xs:string">Товар №1</m:Value>
        <m:ValueType>string</m:ValueType>
        <m:Properties>
          <m:Key>Party</m:Key>
          <m:Value xsi:type="xs:string">d624e9b9-4076-11e3-954c-001e67086478</m:Value>
          <m:ValueType>string</m:ValueType>
          <m:Fields>
            <m:Key>Name</m:Key>
            <m:Value xsi:type="xs:string">29.10.2013</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Fields>
        </m:Properties>
        <m:Fields>
          <m:Key>FullName</m:Key>
          <m:Value xsi:type="xs:string"/>
          <m:ValueType>string</m:ValueType>
        </m:Fields>
        <m:Fields>
          <m:Key>ID</m:Key>
          <m:Value xsi:type="xs:string">#T#000000000000000000274</m:Value>
          <m:ValueType>string</m:ValueType>
        </m:Fields>
        <m:Fields>
          <m:Key>Article</m:Key>
          <m:Value xsi:type="xs:string">M053115302W</m:Value>
          <m:ValueType>string</m:ValueType>
        </m:Fields>
        <m:Fields>
          <m:Key>Balance</m:Key>
          <m:Value xsi:type="xs:decimal">4</m:Value>
          <m:ValueType>float</m:ValueType>
        </m:Fields>
        <m:Fields>
          <m:Key>Reserve</m:Key>
          <m:Value xsi:type="xs:decimal">0</m:Value>
          <m:ValueType>float</m:ValueType>
        </m:Fields>
      </m:List>
      ...
      <m:List>
        <m:Key>a99e2812-fe93-11e2-9e83-001e670c26ac</m:Key>
        <m:Value xsi:type="xs:string">Товар №35</m:Value>
```



```

<m:ValueType>string</m:ValueType>
<m:Properties>
  <m:Key>Party</m:Key>
  <m:Value xsi:type="xs:string">a2bfa079-3695-11e3-9a4c-001e670c42dc</m:Value>
  <m:ValueType>string</m:ValueType>
  <m:Fields>
    <m:Key>Name</m:Key>
    <m:Value xsi:type="xs:string">16.10.2013</m:Value>
    <m:ValueType>string</m:ValueType>
  </m:Fields>
</m:Properties>
<m:Fields>
  <m:Key>FullName</m:Key>
  <m:Value xsi:type="xs:string"/>
  <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>ID</m:Key>
  <m:Value xsi:type="xs:string">#T#00000000000000000162</m:Value>
  <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>Article</m:Key>
  <m:Value xsi:type="xs:string">M074002300W</m:Value>
  <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>Balance</m:Key>
  <m:Value xsi:type="xs:decimal">1</m:Value>
  <m:ValueType>float</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>Reserve</m:Key>
  <m:Value xsi:type="xs:decimal">0</m:Value>
  <m:ValueType>float</m:ValueType>
</m:Fields>
</m>List>
...
...
<m>List>
  <m:Key>078c83ce-fe8a-11e2-b765-001e67086478</m:Key>
  <m:Value xsi:type="xs:string">Товар №xxx</m:Value>
  <m:ValueType>string</m:ValueType>
  <m:Properties>
    <m:Key>Party</m:Key>
    <m:Value xsi:type="xs:string">57ded4e5-882e-11e3-8e77-001e670c26ac</m:Value>
    <m:ValueType>string</m:ValueType>
    <m:Fields>
      <m:Key>Name</m:Key>
      <m:Value xsi:type="xs:string">28.01.2014</m:Value>
      <m:ValueType>string</m:ValueType>
    </m:Fields>
  </m:Properties>
  <m:Fields>
    <m:Key>FullName</m:Key>
    <m:Value xsi:type="xs:string"/>
    <m:ValueType>string</m:ValueType>
  </m:Fields>
  <m:Fields>
    <m:Key>ID</m:Key>
    <m:Value xsi:type="xs:string">#T#00000000000000000094</m:Value>
    <m:ValueType>string</m:ValueType>
  </m:Fields>
  <m:Fields>
    <m:Key>Article</m:Key>
    <m:Value xsi:type="xs:string">M064904500W</m:Value>
    <m:ValueType>string</m:ValueType>
  </m:Fields>
  <m:Fields>
    <m:Key>Balance</m:Key>
    <m:Value xsi:type="xs:decimal">1</m:Value>
    <m:ValueType>float</m:ValueType>
  </m:Fields>
  <m:Fields>
    <m:Key>Reserve</m:Key>
    <m:Value xsi:type="xs:decimal">0</m:Value>
    <m:ValueType>float</m:ValueType>
  </m:Fields>

```




```
        </m:Fields>
      </m:List>
    </m:return>
  </m:GetReferenceDataResponse></soap:Body>
</soap:Envelope>
```



GETREFERENCEDATA: SERVICES

The function allows to get a list of rendered services that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «Services»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «Services» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Global Unique Identifier (GUID) of the reference element
Value	string	Yes	Service name
Fields	Element - list		Other properties of the reference element

Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
Default	boolean		True if this service is used by default



GETREFERENCEDATA: WAREHOUSES

The function allows to get a list of warehouses that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «Repository»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «Repository» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Id of the warehouse
Value	string	Yes	Global Unique Identifier (GUID) of the reference element
Fields	Element - list		Other properties of the reference element
	string		NameStore – warehouse name
	string		GuidGeography – GUID of the geography reference
	string		NameGeography – geography name



GETREFERENCEDATA: CHARACTERISTICS OF THE CLIENT GOODS

The function allows to get a list of client's goods characteristics that are used in other functions called by the web service.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «CharacteristicsOfItems»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «CharacteristicsOfItems» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Global Unique Identifier (GUID) of the reference element
Value	string	Yes	Goods characteristics name
Fields	Element - list		Other properties of the reference element

Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
FullName	string		Full name of the goods characteristic



GETREFERENCEDATA: CARGO STATE

The function allows to get a list of available cargo states.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
Reference	string	Yes	Reference name must be equal «CargoStates»

ANSWER:

Element

«Element» has two properties: the property «**Key**» contains the value «CargoStates» and the property «**List**» contains reference elements. The property «**Properties**» contains the error information (see the chapter [Error information](#)).

Every reference element of the list «**List**» has the following properties :

Property	Type	Required	Description
Key	string	Yes	Global Unique Identifier (GUID) of the reference element
Value	string	Yes	Cargo state name
Fields	Element - list		Other properties of the reference element

Properties values passed through «**Fields**»:

Parameter (Key)	Value type	Required	Description
IsFolder	boolean		True if this element is a folder
Parent	string		Global Unique Identifier (GUID) of the reference parent element
Level	string		Reference level



SAVEDOCUMENTS: ORDERS

The function creates new documents in the LMS «Cargo» or updates existing documents by number.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	data	list of saved documents
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
DocumentType	string	Yes	Document type must be «Order» or «Orders»
StoreDependsOn Destination	boolean		True if the function selects the nearest office on the delivery's geography. Otherwise, it returns the main warehouse of the client. But if the client has only one warehouse in the reference then this warehouse will be selected. If the client has several warehouses in the reference then it is necessary to put the parameter «Repository»

PARAMETER DESCRIPTION DATA

The structure «data» can contains the data of one or several documents as a list of structures «Element». The structure «data» must have the property «Key» that contains the value «Documents». Parameters are put into a list using the property «List» that cannot be empty.

Every document from the «List» has the following properties:

Property	Type	Required	Description
Key	string	Yes	It always must be equal «Order»
Fields	Element - list	Yes	Document attributes
Tables	Element - list		Tables of the document data: description of the composite cargo, goods description for Internet shop's orders

The property «Fields» has the following order's properties:

Parameter (Key)	Value type	Required	Description
ClientNumber	string		Client number of the document (if it exists)
AgentNumber	string		Agent number of the document (if it exists)
ContactPerson	string		GUID (Globally Unique Identifier) of the contact person (how to get contact persons read in the chapter GetReferenceData: client's contact persons)
Department	string		GUID (Globally Unique Identifier) of the department (how to get the departments list read in the chapter GetReferenceData: departments) or department name.



Project	string		GUID (Globally Unique Identifier) of the project (how to get the projects list read in the chapter GetReferenceData: projects)
TakeDate	dateTime		Date of pick up the cargo
TakeTime	string		Time of pick up the cargo, free format
TakeAtOffice	boolean		True if the cargo is accepted at the office to deliver. These values is default by all orders with goods (orders to the warehouse). Otherwise for other orders. That's mean a courier will pick up the cargo from the sender.
DeliveryDate	dateTime		Planned delivery date
DeliveryTime	string		Planned delivery time, free format
Comment	string		Order's comment
Sender	string	Yes	Cargo's sender
SenderOfficial	string		Contact person of the sender
SenderGeography	string	Yes	GUID (Globally Unique Identifier) of the sender's geography (how to get geographical objects read in the chapter GetReferenceData: geography) or client's geography code if the client uses his own reference
SenderAddress	string	Yes	Address where the cargo is picked up
SenderPhone	string	Yes	Phone number of the sender
SenderEMail	string		Sender's e-mail
SenderInfo	string		Extra information of the sender
Recipient	string	Yes	Cargo's recipient
RecipientOfficial	string		Contact person of the recipient
RecipientGeography	string	Yes	GUID (Globally Unique Identifier) of the delivery geography (how to get geographical objects read in the chapter GetReferenceData: geography) or the client's geography code if the client uses his own reference
RecipientAddress	string	Yes	Delivery address
RecipientPhone	string	Yes	Phone number of the recipient
RecipientEMail	string		Recipient's e-mail
RecipientInfo	string		Extra information of the recipient
Repository	String		GUID (Globally Unique Identifier) of the warehouse (how to get a list of warehouses read in the chapter GetReferenceData: warehouses)
Urgency	string	Yes	GUID (Globally Unique Identifier) of the delivery urgency (how to get a list of urgencies read in the chapter GetReferenceData: urgencies)



Payer	float	Yes	Payer's code (how to get a list of payers read in the chapter GetReferenceData: payers)
PaymentMethod	float	Yes	Code of the payment method (how to get a list of payment methods read in the chapter GetReferenceData: payment methods)
ShippingMethod	string	Yes	GUID (Globally Unique Identifier) of the delivery method (how to get a list of delivery methods read in the chapter GetReferenceData: delivery methods)
TypeOfCargo	string	Yes	GUID (Globally Unique Identifier) of the cargo type (how to get a list of cargo types read in the chapter GetReferenceData: cargo types)
WithReturn	boolean		True if after delivery it is necessary to return the cargo to the sender
Weight	float	Yes	Cargo weight, kg
Height	float		Height, cm
Length	float		Length, cm
Width	float		Width, cm
CargoDescription	string		Cargo description
CargoPackageQty	float	Yes	Number of places, non-negative integer
InsuranceRate	float		Insurance value of the cargo
InsuranceRate Currency	string		GUID (Globally Unique Identifier) of the cargo's insurance value (how to get a list of currencies read in the chapter GetReferenceData: currencies); default value is RUB
DeclaredValueRate	float		Declared value of the cargo
DeclaredValueRate Currency	string		GUID (Globally Unique Identifier) of the declared value's currency (how to get a list of currencies read in the chapter GetReferenceData: currencies); default value is RUB
ValueForCustoms Purposes	float		Customs value of the cargo
ValueForCustoms PurposesCurrency	string		GUID (Globally Unique Identifier) of the customs value's currency (how to get a list of currencies read in the chapter GetReferenceData: currencies); default value is RUB
COD	float		Cash on delivery (COD)
CODPayer	float		Code of the COD payer (how to get a list of payers read in the chapter GetReferenceData: payers)
CODDescription	string		Extra information of the COD



CODCurrency	string	GUID (Globally Unique Identifier) of the COD currency (how to get a list of currencies read in the chapter GetReferenceData: currencies); default value is RUB
SenderShippingCost	float	Services cost
ItemsProcessing Action	string	Type of the goods' processing (it makes a sense if the order has a list of goods). It can be one of the following values: «inventory», «assembly», «shipping», «selection» and «incoming»
ReplyEMail	string	E-mail for notifications about the delivery state
ReplySMSPhone	string	Phone number for sms notifications about the delivery state

The property «Tables» has the following parameters list of order's tables:

Property	Type	Required	Description
Key	string	Yes	Table name. It can be: «CargoPackages» - to describe characteristics of the composite cargo; «Items» - to describe the information about the goods list
List	Element - list	Yes	List of table rows. Each cell data of each rows are stored in the property «Fields»

The property «Fields» has the following table fields of the composite cargo («CargoPackages»):

Parameter (Key)	Value type	Required	Description
PackageID	string		Package number. If it isn't empty then it is checked for uniqueness
Weight	float		Cargo weight, kg
Height	float		Height, cm
Length	float		Length, cm
Width	float		Width, cm
CargoPackageQty	float		Number of places, non-negative integer

The property «Fields» has the following table fields of goods («Items»):

Parameter (Key)	Value type	Required	Description
Item	string	Yes	GUID (Globally Unique Identifier) of goods (how to get a list of goods read in the chapter GetReferenceData: goods or GetReferenceData: goods, detail information)
Party	string		GUID (Globally Unique Identifier) of goods batches (how to get a list of batches read in the chapter GetReferenceData: client's goods batches)
SerialNumber	string		GUID (Globally Unique Identifier) of goods serial

numbers (how to get a list of serial numbers read in the chapter [GetReferenceData: serial numbers of client's goods](#))

Package	string		GUID (Globally Unique Identifier) of measurement units, packages (how to get a list of measurement units read in the chapter GetReferenceData: measurement units)
VATRate	float		Code of VAT rate (how to get a list of VAT rates read in the chapter GetReferenceData: VAT rates)
Qty	float	Yes	Quantity, non-negative integer
Price	float		Price
Barcode	string		Producer's barcode

ANSWER:

[Element](#)

«Element» has two properties: the property «Key» contains the value «SaveDocuments» and the property «List» contains the document information. The property «Properties» contains the error information (see the chapter [Error information](#)).

Every answer of the list «List» contains the following properties:

Property	Type	Required	Description
Key	string	Yes	It always must be equal «Order»
Properties	Element - list	Yes	Document property or error information of the document (see the chapter Error information)

The list of properties «**Properties**»:

Parameter (Key)	Value type	Required	Description
Number	string	Yes	Document number
ClientNumber	string		Client number of the document (if it exists)
AgentNumber	string		Agent number of the document (if it exists)
CreateDate	dateTime	Yes	Document date
DeliveryDate	dateTime		Planned delivery date



EXAMPLE:

Request:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body> <m:SaveDocuments xmlns:m="http://www.cargo3.ru">
  <m:login xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">LOGIN</m:login>
  <m:password xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">PASSWORD</m:password>
  <m:data xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <m:Key>Orders</m:Key>
    <m>List>
      <m:Key>Order</m:Key>
      <m:Properties>
        <m:Key>ClientNumber</m:Key>
        <m:Value xsi:type="xs:string">123</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Properties>
      <m:Fields>
        <m:Key>Department</m:Key>
        <m:Value xsi:type="xs:string"/>
        <m:ValueType>string</m:ValueType>
      </m:Fields>
      <m:Fields>
        <m:Key>Project</m:Key>
        <m:Value xsi:type="xs:string"/>
        <m:ValueType>string</m:ValueType>
      </m:Fields>
      <m:Fields>
        <m:Key>TakeDate</m:Key>
        <m:Value xsi:type="xs:dateTime">2014-04-15T00:00:00</m:Value>
        <m:ValueType>dateTime</m:ValueType>
      </m:Fields>
      <m:Fields>
        <m:Key>TakeTime</m:Key>
        <m:Value xsi:type="xs:string">C 10 да 12</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Fields>
      <m:Fields>
        <m:Key>Comment</m:Key>
        <m:Value xsi:type="xs:string">Комментарий общий</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Fields>
      <m:Fields>
        <m:Key>Sender</m:Key>
        <m:Value xsi:type="xs:string">Отправитель</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Fields>
      <m:Fields>
        <m:Key>SenderOfficial</m:Key>
        <m:Value xsi:type="xs:string">Контактное лицо отправителя</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Fields>
      <m:Fields>
        <m:Key>SenderGeography</m:Key>
        <m:Value xsi:type="xs:string">cf862f56-442d-11dc-9497-0015170f8c09</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Fields>
      <m:Fields>
        <m:Key>SenderAddress</m:Key>
        <m:Value xsi:type="xs:string">Адрес отправителя</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Fields>
      <m:Fields>
        <m:Key>SenderPhone</m:Key>
        <m:Value xsi:type="xs:string">123-45-67</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Fields>
      <m:Fields>
        <m:Key>SenderEMail</m:Key>
        <m:Value xsi:type="xs:string">sender@gmail.com</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Fields>
    </m:List>
  </m:data>
  </m:SaveDocuments>
</soap:Body>
</soap:Envelope>
```



```

        <m:Key>SenderInfo</m:Key>
        <m:Value xsi:type="xs:string">Комментарий отправителя</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Recipient</m:Key>
        <m:Value xsi:type="xs:string">Получатель</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>RecipientOfficial</m:Key>
        <m:Value xsi:type="xs:string">Контактное лицо получателя</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>RecipientGeography</m:Key>
        <m:Value xsi:type="xs:string">cf862f77-442d-11dc-9497-0015170f8c09</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>RecipientAddress</m:Key>
        <m:Value xsi:type="xs:string">Адрес получателя</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>RecipientPhone</m:Key>
        <m:Value xsi:type="xs:string">765-43-21</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>RecipientEMail</m:Key>
        <m:Value xsi:type="xs:string">recipient@gmail.com</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>RecipientInfo</m:Key>
        <m:Value xsi:type="xs:string">Комментарий получателя</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Urgency</m:Key>
        <m:Value xsi:type="xs:string">18c4f207-458b-11dc-9497-0015170f8c09</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Payer</m:Key>
        <m:Value xsi:type="xs:string">2</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>PaymentMethod</m:Key>
        <m:Value xsi:type="xs:string">0</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>TypeOfCargo</m:Key>
        <m:Value xsi:type="xs:string">4aab1fc6-fc2b-473a-8728-58bcd4ff79ba</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Weight</m:Key>
        <m:Value xsi:type="xs:decimal">2</m:Value>
        <m:ValueType>float</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>CargoDescription</m:Key>
        <m:Value xsi:type="xs:string">Описание груза</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>CargoPackageQty</m:Key>
        <m:Value xsi:type="xs:decimal">2</m:Value>
        <m:ValueType>int</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>DeliveryDate</m:Key>
        <m:Value xsi:type="xs:dateTime">2014-04-17T00:00:00</m:Value>

```



```

        <m:ValueType>dateTime</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>ItemsProcessingAction</m:Key>
        <m:Value xsi:type="xs:string">incoming</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Tables>
        <m:Key>Items</m:Key>
        <m>List>
            <m:Key>ItemData</m:Key>
            <m:Fields>
                <m:Key>Item</m:Key>
                <m:Value xsi:type="xs:string">article-M055001011W </m:Value>
                <m:ValueType>string</m:ValueType>
            </m:Fields>
            <m:Fields>
                <m:Key>Qty</m:Key>
                <m:Value xsi:type="xs:decimal">1</m:Value>
                <m:ValueType>int</m:ValueType>
            </m:Fields>
            <m:Fields>
                <m:Key>Price</m:Key>
                <m:Value xsi:type="xs:decimal">500</m:Value>
                <m:ValueType>float</m:ValueType>
            </m:Fields>
            <m:Fields>
                <m:Key>VATRate</m:Key>
                <m:Value xsi:type="xs:string">0 </m:Value>
                <m:ValueType>string</m:ValueType>
            </m:Fields>
        </m>List>
        <m>List>
            <m:Key>ItemData</m:Key>
            <m:Fields>
                <m:Key>Item</m:Key>
                <m:Value xsi:type="xs:string">article-M082701000W </m:Value>
                <m:ValueType>string</m:ValueType>
            </m:Fields>
            <m:Fields>
                <m:Key>Qty</m:Key>
                <m:Value xsi:type="xs:decimal">1</m:Value>
                <m:ValueType>int</m:ValueType>
            </m:Fields>
            <m:Fields>
                <m:Key>Price</m:Key>
                <m:Value xsi:type="xs:decimal">1500</m:Value>
                <m:ValueType>float</m:ValueType>
            </m:Fields>
            <m:Fields>
                <m:Key>VATRate</m:Key>
                <m:Value xsi:type="xs:string">5 </m:Value>
                <m:ValueType>string</m:ValueType>
            </m:Fields>
        </m>List>
    </m:Tables>
</m>List>
</m:data>
<m:parameters xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <m:Key>Parameters</m:Key>
    <m>List>
        <m:Key>DocumentType</m:Key>
        <m:Value xsi:type="xs:string">order</m:Value>
        <m:ValueType>string</m:ValueType>
    </m>List>
</m:parameters>
</m:SaveDocuments></soap:Body>
</soap:Envelope>

```



Answer:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body> <m:SaveDocumentsResponse xmlns:m="http://www.cargo3.ru">
    <m:return xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <m:Key>SaveDocuments</m:Key>
      <m:List>
        <m:Key>Order</m:Key>
        <m:Properties>
          <m:Key>Number</m:Key>
          <m:Value xsi:type="xs:string">999-0001356552</m:Value>
          <m:ValueType>string</m:ValueType>
        </m:Properties>
        <m:Properties>
          <m:Key>CreateDate</m:Key>
          <m:Value xsi:type="xs:dateTime">2014-04-14T00:25:14</m:Value>
          <m:ValueType>dateTime</m:ValueType>
        </m:Properties>
        <m:Properties>
          <m:Key>DeliveryDate</m:Key>
          <m:Value xsi:type="xs:dateTime">2014-04-15T00:00:00</m:Value>
          <m:ValueType>dateTime</m:ValueType>
        </m:Properties>
      </m:List>
    </m:return>
  </m:SaveDocumentsResponse></soap:Body>
</soap:Envelope>
```



TRACKING: TRACKING OF CARGO STATE

The function allows to get the history about cargo states and the current state on the order or waybill number.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	documents	document list for which we request information
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
DocumentType	string		Type of the document tracking. It can be «Order»/«Orders» or «Waybill»/«Waybills»
NumberType	string		Number type is used for the document search. If number type is empty then the search will be carried out by the internal number. Available values: «ClientNumber» (client number), «AgentNumber» (agent number), «OrderNumber» (order number, it is used only for the waybill's search) and «PackageID» (id of the package)
OnlySelectedType	boolean		True if the search is carried out strictly by the number of NumberType . If it is false and the function cannot find any document by the number of NumberType then the search will be carried out by the internal number from the LMS «Cargo».

PARAMETER DESCRIPTION DATA

The structure «documents» can contain the data of one or several documents as a list of structures «Element». The structure «documents» must have the property «Key» that contains the value «Documents». Documents numbers are put into a list using the property «List» that cannot be empty.

Every document from the «List» has the following properties:

Property	Type	Required	Description
Key	string	Yes	Number of the tracked document
Properties	Element - list		Additional properties. They are completely the same as passed in the function's parameter « parameters » and define rules to search document. If one of these properties is empty then general properties from « parameters » will be used to search document.

ANSWER:

Element

«Element» has two properties: the property «Key» contains the value «Tracking» and the property «List» contains information of documents states. The property «Properties» contains the error information (see the chapter [Error information](#)).

Every element of the list «List» has information of the document and contains the following properties :



Property	Type	Required	Description
Key	string	Yes	Document number in the LMS «Cargo»
Value	string	Yes	Document type: «Order» or «Waybill»
Properties	Element - list	Yes	Extra information of the document
List	Element - list	Yes	History of document states

Properties list in «**Properties**»:

Parameter (Key)	Value type	Required	Description
DocumentType	string	Yes	Document type: «Order» or «Waybill»
DocumentName	string	Yes	Internal document name in the LMS «Cargo»
GUID	string		GUID (Globally Unique Identifier) of the document in the LMS «Cargo»
Date	dateTime		Document date and time
Number	string		Document number in the LMS «Cargo»
ClientNumber	string		Client number of the document (if it exists)
Posted	boolean		True if this document not only is written in the LMS «Cargo» but also is accountable

The property «List» contains all available states as a structure «Element». Every element contains the following properties:

Property	Type	Required	Description
Key	string	Yes	State description
Value	string	Yes	State note
Properties	Element - list	Yes	Various state properties



Additional information of the state in the property «**Properties**» has the following parameters:

Parameter (Key)	Value type	Required	Description
GUID	string	Yes	GUID (Globally Unique Identifier) of the state in the LMS «Cargo»
Comment	string		State note
RecorderGUID	string	Yes	GUID (Globally Unique Identifier) of the document in the LMS «Cargo»
RecorderName	string	Yes	Internal document name in the LMS «Cargo»
DateTime	dateTime	Yes	Recording date and time of the state
DeliveryDateTime	dateTime		Delivery date and time (only for state when the cargo is delivered)
Recipient	string		Cargo recipient (only for state when the cargo is delivered)



EXAMPLE:

Request:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body> <m:Tracking xmlns:m="http://www.cargo3.ru">
    <m:login xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">LOGIN</m:login>
    <m:password xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">PASSWORD</m:password>
    <m:documents xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <m:Key>Documents</m:Key>
      <m:Properties>
        <m:Key>DocumentType</m:Key>
        <m:Value xsi:type="xs:string">Order</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Properties>
      <m>List>
        <m:Key>495-0007058175</m:Key>
      </m>List>
    </m:documents>
    <m:parameters xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <m:Key>Parameters</m:Key>
    </m:parameters>
  </m:Tracking></soap:Body>
</soap:Envelope>
```

Answer:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body> <m:TrackingResponse xmlns:m="http://www.cargo3.ru">
    <m:return xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <m:Key>Tracking</m:Key>
      <m>List>
        <m:Key>495-0007058175</m:Key>
        <m:Value xsi:type="xs:string">Order</m:Value>
        <m:ValueType>string</m:ValueType>
        <m:Properties>
          <m:Key>DocumentType</m:Key>
          <m:Value xsi:type="xs:string">Order</m:Value>
          <m:ValueType>string</m:ValueType>
        </m:Properties>
        <m:Properties>
          <m:Key>DocumentName</m:Key>
          <m:Value xsi:type="xs:string">Заказ</m:Value>
          <m:ValueType>string</m:ValueType>
        </m:Properties>
        <m:Properties>
          <m:Key>GUID</m:Key>
          <m:Value xsi:type="xs:string">f837e5b9-bbf9-11e3-810e-001e670c42dc</m:Value>
          <m:ValueType>string</m:ValueType>
        </m:Properties>
        <m:Properties>
          <m:Key>Date</m:Key>
          <m:Value xsi:type="xs:dateTime">2014-04-04T16:37:19</m:Value>
          <m:ValueType>dateTime</m:ValueType>
        </m:Properties>
        <m:Properties>
          <m:Key>Number</m:Key>
          <m:Value xsi:type="xs:string">495-0007058175</m:Value>
          <m:ValueType>string</m:ValueType>
        </m:Properties>
      </m>List>
      <m:Key>Заказ принят, идет обработка заказа.</m:Key>
      <m:Value xsi:type="xs:string"/>
      <m:ValueType>string</m:ValueType>
      <m:Properties>
        <m:Key>GUID</m:Key>
        <m:Value xsi:type="xs:string">0bcffda3-d0ca-4104-bc58-fbd374f325cd</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Properties>
    </m:return>
  </m:TrackingResponse>
</soap:Body>
</soap:Envelope>
```



```

    <m:Properties>
      <m:Key>Comment</m:Key>
      <m:Value xsi:type="xs:string"/>
      <m:ValueType>string</m:ValueType>
    </m:Properties>
  </m:List>
  <m:List>
    <m:Properties>
      <m:Key>RecorderGUID</m:Key>
      <m:Value xsi:type="xs:string">f837e5b9-bbf9-11e3-810e-001e670c42dc</m:Value>
      <m:ValueType>string</m:ValueType>
    </m:Properties>
    <m:Properties>
      <m:Key>RecorderName</m:Key>
      <m:Value xsi:type="xs:string">Заказ</m:Value>
      <m:ValueType>string</m:ValueType>
    </m:Properties>
    <m:Properties>
      <m:Key>DateTime</m:Key>
      <m:Value xsi:type="xs:dateTime">2014-04-04T16:37:19</m:Value>
      <m:ValueType>dateTime</m:ValueType>
    </m:Properties>
  </m:List>
  <m:List>
    <m:Key>Отправление доставлено получателю.</m:Key>
    <m:Value xsi:type="xs:string"/>
    <m:ValueType>string</m:ValueType>
    <m:Properties>
      <m:Key>GUID</m:Key>
      <m:Value xsi:type="xs:string">052bdc4e-02b2-4209-b2ab-1e856d330ef5</m:Value>
      <m:ValueType>string</m:ValueType>
    </m:Properties>
    <m:Properties>
      <m:Key>Comment</m:Key>
      <m:Value xsi:type="xs:string"/>
      <m:ValueType>string</m:ValueType>
    </m:Properties>
    <m:Properties>
      <m:Key>RecorderGUID</m:Key>
      <m:Value xsi:type="xs:string">352691af-bbfa-11e3-810e-001e670c42dc</m:Value>
      <m:ValueType>string</m:ValueType>
    </m:Properties>
    <m:Properties>
      <m:Key>RecorderName</m:Key>
      <m:Value xsi:type="xs:string">ЗаказРегистрацияСостояния</m:Value>
      <m:ValueType>string</m:ValueType>
    </m:Properties>
    <m:Properties>
      <m:Key>DateTime</m:Key>
      <m:Value xsi:type="xs:dateTime">2014-04-04T17:08:30</m:Value>
      <m:ValueType>dateTime</m:ValueType>
    </m:Properties>
  </m:List>
</m:List>
</m:return>
</m:TrackingResponse></soap:Body>
</soap:Envelope>

```



PRINTDOCUMENT: PRINT DOCUMENT

Print the document.

REQUEST PARAMETERS:

string	login	user name
string	password	password
string	type	document type (order or waybill)
string	printFormName	name of print form (ПечатнаяФормаНакладнойДляЗаказа – for order, ПечатнаяФормаНакладной – for waybill)
string	number	document number
string	parametrs	additional parameters

ANSWER:

ResultPrintArray

Parameter	Type	Required	Description
Error	boolean	Yes	True if there was an error while printing document
ErrorInfo	string	No	Error description
Items	resultprint	No	Array of function results

DESCRIPTION OF AUXILIARY TYPE **RESULTPRINT**

Parameters:

Parameter	Type	Required	Description
Error	boolean	Yes	True if it is an error
ErrorInfo	string	No	Error description
Value	string	No	Binary data of the print form
Name	string	No	File name
Size	string	No	File size



REPORT: DETAILED BILLS

The function allows to get the report «Detailed bills» that contains period detail information about bills of rendered services and processed waybills.

REQUEST PARAMETERS:

string	login	user name
string	password	password
string	name	report name, it must be equal «SummaryOfPayments»
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
FromDate	dateTime		The report selects data from this date. If it is empty, the report will select the data from the beginning of last month.
ToDate	dateTime		The report selects data up to this date. If it is empty, the report will select the data up to the end of last month.
OrderNumber	string		Order number. Put it if you want to generate the report only for one order
WaybillNumber	string		Waybill number. Put it if you want to generate the report only for one waybill.

ANSWER:

Element

«Element» has three properties: the property «Key» contains the value «Report», the property «Fields» contains a list of report fields in the order they are output in a report structure and the property «List» contains the data report. The property «Properties» contains the error information (see the chapter [Error information](#)).

Every field description from the «Fields» has the following properties:

Property	Type	Required	Description
Key	string	Yes	Name of the report's column
Value	string	Yes	Data type of the report's column



The report has the following columns (check columns set with answer's list of «Fields» because it can change):

Column	Type	Description
Return	boolean	True if return the delivery
Readdress	boolean	True if readdress the delivery
WaybillNumber	string	Waybill's number
OrderNumber	string	Order's number
PickupDate	dateTime	Date of pick up the cargo
ContactPerson	string	Contact person
Origin	string	Departure place (human settlement or country)
Destination	string	Destination (human settlement or country)
Urgency	string	Urgency
TypeOfCargo	string	Cargo type
DeliveryDate	dateTime	Delivery date
RecipientPerson	string	Person who received a cargo
Qty	int	Number of places
Weight	float	Weight, kg.
Price	float	Delivery cost

Every element from the «List» contains description of the document «Waybills grouping» as the structure «Element» and has the following properties:

Property	Type	Required	Description
Key	string	Yes	«SummaryOfWaybills»
Value	string	Yes	Number of the waybills grouping
Properties	Element - list		Additional properties of the waybills grouping
Fields	Element - list		Additional fields of the waybills grouping
List	Element - list		Rendered services' list of this waybills grouping

The list of properties in «Properties» for the document «Waybills grouping»:

Parameter (Key)	Value type	Required	Description
Number	string	Yes	Number of the document «Waybills grouping»
Date	dateTime	Yes	Date of the document «Waybills grouping»
GUID	string		GUID (Globally Unique Identifier) of the document in the LMS «Cargo»



The list of properties in «**Fields**» for the document «Waybills grouping»:

Parameter (Key)	Value type	Required	Description
Client	string	Yes	Client
Contract	string	Yes	Contract name (number)
ContractDate	dateTime	Yes	Contract date
ContractCurrency	string	Yes	Contract currency
Comment	string	Yes	Contract comment
Total	float	Yes	Total amount of the document «Waybills grouping»
TotalText	string	Yes	Text description (value) of the field « Total »

Every element from the «List» contains description of rendered service as the structure «Element» and has the following properties:

Property	Type	Required	Description
Key	string	Yes	«ServiceName»
Value	string	Yes	Service name
Properties	Element - list		Additional properties of the service
Rows	Element - list		Report lines. Every line contains the waybill from the document «Waybills grouping»

The properties list of the service in «**Properties**»:

Parameter (Key)	Value type	Required	Description
Total	float	Yes	Total cost of all deliveries on this service
TotalWithRebate	float		Total cost of all deliveries on this service with discount
DiscountPercent	float		Discount percent
GUID	string	Yes	GUID (Globally Unique Identifier) of the service in the LMS «Cargo»

Every element from the list «**Rows**» in the structure «Element» contains the waybill's data from the document «Waybill grouping» and is represented as the structure «Row». The waybill's data is in the list «Cells» of the structure «Row». The data content and the order in «Cells» is a list of columns that are described above.



EXAMPLE:

Request:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body> <m:Report xmlns:m="http://www.cargo3.ru">
    <m:login xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">LOGIN</m:login>
    <m:password xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">PASSWORD</m:password>
    <m:name xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">SummaryOfPayments</m:name>
    <m:parameters xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <m:Key>Parameters</m:Key>
      <m>List>
        <m:Key>FromDate</m:Key>
        <m:Value xsi:type="xs:dateTime">2014-09-01T00:00:00</m:Value>
        <m:ValueType>dateTime</m:ValueType>
      </m>List>
      <m>List>
        <m:Key>ToDate</m:Key>
        <m:Value xsi:type="xs:dateTime">2014-09-30T00:00:00</m:Value>
        <m:ValueType>dateTime</m:ValueType>
      </m>List>
    </m:parameters>
  </m:Report></soap:Body>
</soap:Envelope>
```

Answer:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body> <m:ReportResponse xmlns:m="http://www.cargo3.ru">
    <m:return xmlns:xs="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <m:Key>Report</m:Key>
      <m:Fields>
        <m:Key>Return</m:Key>
        <m:Value xsi:type="xs:string">boolean</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Fields>
      <m:Fields>
        <m:Key>Readdress</m:Key>
        <m:Value xsi:type="xs:string">boolean</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Fields>
      <m:Fields>
        <m:Key>waybillNumber</m:Key>
        <m:Value xsi:type="xs:string">string</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Fields>
      <m:Fields>
        <m:Key>OrderNumber</m:Key>
        <m:Value xsi:type="xs:string">string</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Fields>
      <m:Fields>
        <m:Key>PickupDate</m:Key>
        <m:Value xsi:type="xs:string">dateTime</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Fields>
      <m:Fields>
        <m:Key>ContactPerson</m:Key>
        <m:Value xsi:type="xs:string">string</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Fields>
      <m:Fields>
        <m:Key>Origin</m:Key>
        <m:Value xsi:type="xs:string">string</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Fields>
      <m:Fields>
        <m:Key>Destination</m:Key>
        <m:Value xsi:type="xs:string">string</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Fields>
      <m:Fields>
        <m:Key>Urgency</m:Key>
        <m:Value xsi:type="xs:string">string</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Fields>
      <m:Fields>
        <m:Key>TypeOfCargo</m:Key>
        <m:Value xsi:type="xs:string">string</m:Value>
      </m:Fields>
    </m:return>
  </m:ReportResponse>
</soap:Body>
</soap:Envelope>
```




```

        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>DeliveryDate</m:Key>
        <m:Value xsi:type="xs:string">dateTime</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>RecipientPerson</m:Key>
        <m:Value xsi:type="xs:string">string</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Qty</m:Key>
        <m:Value xsi:type="xs:string">int</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Weight</m:Key>
        <m:Value xsi:type="xs:string">float</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Price</m:Key>
        <m:Value xsi:type="xs:string">float</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m>List>
        <m:Key>SummaryOfwaybills</m:Key>
        <m:Value xsi:type="xs:string">498-00000118294</m:Value>
        <m:ValueType>string</m:ValueType>
        <m:Properties>
            <m:Key>Number</m:Key>
            <m:Value xsi:type="xs:string">498-00000118294</m:Value>
            <m:ValueType>string</m:ValueType>
        </m:Properties>
        <m:Properties>
            <m:Key>Date</m:Key>
            <m:Value xsi:type="xs:dateTime">2014-08-31T23:59:59</m:Value>
            <m:ValueType>dateTime</m:ValueType>
        </m:Properties>
        <m:Fields>
            <m:Key>Client</m:Key>
            <m:Value xsi:type="xs:string">CLIENT</m:Value>
            <m:ValueType>string</m:ValueType>
        </m:Fields>
        <m:Fields>
            <m:Key>Contract</m:Key>
            <m:Value xsi:type="xs:string">CONTRACT</m:Value>
            <m:ValueType>string</m:ValueType>
        </m:Fields>
        <m:Fields>
            <m:Key>ContractDate</m:Key>
            <m:Value xsi:type="xs:dateTime">2014-04-01T00:00:00</m:Value>
            <m:ValueType>dateTime</m:ValueType>
        </m:Fields>
        <m:Fields>
            <m:Key>ContractCurrency</m:Key>
            <m:Value xsi:type="xs:string">RUR</m:Value>
            <m:ValueType>string</m:ValueType>
        </m:Fields>
        <m:Fields>
            <m:Key>ContractCurrencyCode</m:Key>
            <m:Value xsi:type="xs:string">643</m:Value>
            <m:ValueType>string</m:ValueType>
        </m:Fields>
        <m:Fields>
            <m:Key>Comment</m:Key>
            <m:Value xsi:type="xs:string"/>
            <m:ValueType>string</m:ValueType>
        </m:Fields>
        <m:Fields>
            <m:Key>Total</m:Key>
            <m:Value xsi:type="xs:decimal">125834.73</m:Value>
            <m:ValueType>float</m:ValueType>
        </m:Fields>
        <m:Fields>
            <m:Key>TotalText</m:Key>
            <m:Value xsi:type="xs:string">Сто двадцать пять тысяч восемьсот тридцать
четыре Рубля 73 копейки</m:Value>
            <m:ValueType>string</m:ValueType>
        </m:Fields>
    </m>List>
    <m:Key>ServiceName</m:Key>
    <m:Value xsi:type="xs:string">Обработка возврата</m:Value>
    <m:ValueType>string</m:ValueType>
    <m:Properties>

```



```

        <m:Key>Total</m:Key>
        <m:Value xsi:type="xs:decimal">400</m:Value>
        <m:ValueType>float</m:ValueType>
    </m:Properties>
    <m:Rows>
        <m:Cells xsi:type="xs:boolean">>false</m:Cells>
        <m:Cells xsi:type="xs:boolean">>false</m:Cells>
        <m:Cells xsi:type="xs:string">495-42469-3805</m:Cells>
        <m:Cells xsi:type="xs:string">495-0007626010</m:Cells>
        <m:Cells xsi:type="xs:dateTime">2014-07-08T00:00:00</m:Cells>
        <m:Cells xsi:type="xs:string"/>
        <m:Cells xsi:type="xs:string">Москва г</m:Cells>
        <m:Cells xsi:type="xs:string">Москва г</m:Cells>
        <m:Cells xsi:type="xs:string">Суточная</m:Cells>
        <m:Cells xsi:type="xs:string">Груз</m:Cells>
        <m:Cells xsi:type="xs:dateTime">2014-08-07T22:19:34</m:Cells>
        <m:Cells xsi:type="xs:string">Возврат</m:Cells>
        <m:Cells xsi:type="xs:decimal">1</m:Cells>
        <m:Cells xsi:type="xs:decimal">4.3</m:Cells>
        <m:Cells xsi:type="xs:decimal">25</m:Cells>
    </m:Rows>
    <m:Rows>
        <m:Cells xsi:type="xs:boolean">>false</m:Cells>
        <m:Cells xsi:type="xs:boolean">>false</m:Cells>
        <m:Cells xsi:type="xs:string">999-0001740356</m:Cells>
        <m:Cells xsi:type="xs:string">999-0001740356</m:Cells>
        <m:Cells xsi:type="xs:dateTime">2014-08-20T00:00:00</m:Cells>
        <m:Cells xsi:type="xs:string"/>
        <m:Cells xsi:type="xs:string">Москва г</m:Cells>
        <m:Cells xsi:type="xs:string">Москва г</m:Cells>
        <m:Cells xsi:type="xs:string">Суточная</m:Cells>
        <m:Cells xsi:type="xs:string">Груз</m:Cells>
        <m:Cells xsi:type="xs:dateTime">2014-08-21T16:40:00</m:Cells>
        <m:Cells xsi:type="xs:string">Маретин</m:Cells>
        <m:Cells xsi:type="xs:decimal">1</m:Cells>
        <m:Cells xsi:type="xs:decimal">0.2</m:Cells>
        <m:Cells xsi:type="xs:decimal">6.65</m:Cells>
    </m:Rows>
    ...
    ...
    ...
    <m:Rows>
        <m:Cells xsi:type="xs:boolean">>false</m:Cells>
        <m:Cells xsi:type="xs:boolean">>false</m:Cells>
        <m:Cells xsi:type="xs:string">495-42469-4048</m:Cells>
        <m:Cells xsi:type="xs:string">495-0007909347</m:Cells>
        <m:Cells xsi:type="xs:dateTime">2014-08-21T00:00:00</m:Cells>
        <m:Cells xsi:type="xs:string"/>
        <m:Cells xsi:type="xs:string">Москва г</m:Cells>
        <m:Cells xsi:type="xs:string">Москва г</m:Cells>
        <m:Cells xsi:type="xs:string">Суточная</m:Cells>
        <m:Cells xsi:type="xs:string">Груз</m:Cells>
        <m:Cells xsi:type="xs:dateTime">2014-08-21T12:00:00</m:Cells>
        <m:Cells xsi:type="xs:string">Тарасов</m:Cells>
        <m:Cells xsi:type="xs:decimal">1</m:Cells>
        <m:Cells xsi:type="xs:decimal">5</m:Cells>
        <m:Cells xsi:type="xs:decimal">100</m:Cells>
    </m:Rows>
</m>List>
<m>List>
    <m:Key>ServiceName</m:Key>
    <m:Value xsi:type="xs:string">Хранение товаров свыше 45 суток</m:Value>
    <m:ValueType>string</m:ValueType>
    <m:Properties>
        <m:Key>Total</m:Key>
        <m:Value xsi:type="xs:decimal">10232.82</m:Value>
        <m:ValueType>float</m:ValueType>
    </m:Properties>
    <m:Rows>
        <m:Cells xsi:type="xs:boolean">>false</m:Cells>
        <m:Cells xsi:type="xs:boolean">>false</m:Cells>
        <m:Cells xsi:type="xs:string">495-0007972445</m:Cells>
        <m:Cells xsi:type="xs:string"/>
        <m:Cells xsi:type="xs:dateTime">2014-08-29T00:00:00</m:Cells>
        <m:Cells xsi:type="xs:string"/>
        <m:Cells xsi:type="xs:string">Москва г</m:Cells>
        <m:Cells xsi:type="xs:string">Москва г</m:Cells>
        <m:Cells xsi:type="xs:string">Срочная</m:Cells>
        <m:Cells xsi:type="xs:string">Груз</m:Cells>
        <m:Cells xsi:nil="true"/>
        <m:Cells xsi:type="xs:string"/>
        <m:Cells xsi:type="xs:decimal">23</m:Cells>
        <m:Cells xsi:type="xs:decimal">0.1</m:Cells>
        <m:Cells xsi:type="xs:decimal">10232.82</m:Cells>
    </m:Rows>
</m>List>
</m>List>

```



```
</m:return>  
</m:ReportResponse></soap:Body>  
</soap:Envelope>
```



SAVEWAYBILLOFFICE: WAYBILLS WITH GOODS

The function creates new waybills in the LMS «Cargo» and creates new waybills with goods for the shipment.

REQUEST PARAMETERS:

string	language	it isn't used
string	login	user name
string	password	password
string	company	it isn't used
string	number	waybill number to edit
string	clientnumber	client document number
Order	orderdata	structure of the document data
string	office	it isn't used

TYPE DESCRIPTION ORDER

Parameters:

Parameter	Type	Required	Description
ClientContact	string	No	Contact person of client
TakeDate	dateTime	No	Date of pick up the waybill
Comment	string	No	Comment of the waybill
Repository	string	No	Warehouse (GUID)
ReplySMSPhone	string	No	Phone number for the sms-informing
ReplyEMail	string	No	E-mail for the notification
Sender	DestinationInformation	Yes	Sender information
Recipient	DestinationInformation	Yes	Recipient information
TypeOfPayer	int	No	Payer, the value by default is a customer
WayOfPayment	int	No	Payment method, the value by default is the non-cash payment
TypeOfCargo	string	No	Cargo type (GUID), the value by default is cargo
Department	string	No	Client's department (GUID)
Project	string	No	Client's project (GUID)
DeliveryOfCargo	int	No	Delivery type (the value by default – to the door, 0 – with cash on delivery, 2 – with return, 3 – with notification, 4 – with return and notification, 5 – customer pickup, 6 – warehouse-door, 7 – warehouse-warehouse)
DeliveryDate	dateTime	No	Planned delivery date
DeliveryTime	string	No	Planned delivery time
ParentOrderForWaybill	string	No	Order or waybill number



TypeOfParentForWaybill	string	No	The document type of the base (order/waybill)
ArrayOfDocumentsPhoto	DocumentsPhoto	No	The document list to make a photo. It uses if the client has the special mode (only for waybills). It mustn't empty if document list deffer from the general rules.

DESCRIPTION OF AUXILIARY TYPE **DESTINATIONINFORMATION**

Parameters:

Parameter	Type	Required	Description
Client	string	Yes	Sender/Recipient
Official	string	No	Contact person of the sender/recipient
Address	destinationAddress	Yes	Address of the sender/recipient
Info	string	No	Comment of the sender/recipient
Phone	string	Yes	Phone number of the sender/recipient
Cargo	Cargo	Yes	Cargo information, only for recipients
EMail	string	No	E-mail of the sender/recipient
Urgency	string	Yes	Urgency (GUID) (this parameter must be passed recipient's block)
Products	Products	No/Yes	Goods information (It is required for Internet shop) Amount of the COD is calculated as total of goods. The payer is always the recipient.

DESCRIPTION OF AUXILIARY TYPE **DESTINATIONADDRESS**

Parameters:

Parameter	Type	Required	Description
Geography	string	Yes	Human settlement (GUID)
Comment	string	No	Extra information about address
FreeForm	boolean	Yes	Always true (true)
Info	string	Yes	Full address



DESCRIPTION OF AUXILIARY TYPE **CARGO**

Parameters:

Parameter	Type	Required	Description
CargoDescription	string	No	Cargo description
CargoPackageQty	string	Yes	Cargo quantity
Weight	float	Yes	Cergo weight
VolumeWeight	float	No	Volume weight
Length	float	No	Length
Width	float	No	Width
Height	float	No	Height
InsuranceRate	float	No	Insurance value
InsuranceRateCurrency	string	No	Currency of the insurance value (the value by default - RUB)
DeclaredValueRate	float	No	Declared value
DeclaredValueRateCurrency	string	No	Currency of the declared value (the value by default - RUB)
PackageID	string	No	Package number
CODPayer	string	No	Payer of the COD (0 – recipient, 1-sender). If COD's payer and recipient are equal then the commodity part of the document mustn't be empty. Only for the Internet shop.

DESCRIPTION OF AUXILIARY TYPE **DOCUMENTSPHOTO**

Parameters:

Parameter	Type	Required	Description
Document	string	No	GUID of the document (the document name if you don't have another settings)
Required	boolean	No	True if this document's photo is required



DESCRIPTION OF AUXILIARY TYPE **PRODUCTS**

Parameters:

Parameter	Type	Required	Description
Article	string	Yes	Goods article
Name	string	No	GUID of goods
Price	float	No	Price per goods unit. The value by default - 0.
PackageQty	int	Yes	Packages quantity
Qty	int	Yes	quantity
VATRate	int	No	VAT rate from the LMS «Cargo». The value by default - 18%.

Pay attention, transfer all goods using in waybill before its creation (functionUpdateClientProducts).

ANSWER:

[ResultArray](#)

Parameter	Type	Required	Description
Error	boolean	Yes	True if it is an error
ErrorInfo	string	No	Error description
Items	resultstring	No	Array of function's results

DESCRIPTION OF AUXILIARY TYPE **RESULTSTRING**

Parameters:

Parameter	Type	Required	Description
Error	boolean	Yes	True if it is an error
ErrorInfo	string	No	Error description
Value	string	No	Function's result



EXAMPLE:

Request:

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <tns:SaveWaybillOffice xmlns:tns="http://www.cargo3.ru"
xmlns="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <tns:Language></tns:Language>
      <tns:Login>ClientNameA_V</tns:Login>
      <tns>Password>*****</tns>Password>
      <tns:Company></tns:Company>
      <tns:Number></tns:Number>
      <tns:ClientNumber>T123456789</tns:ClientNumber>
      <tns:OrderData>
        <tns:ClientContact></tns:ClientContact>
        <tns:Contract></tns:Contract>
        <tns:CreateDate>2015-01-22 06:53:25</tns:CreateDate>
        <tns:DeliveryDate>2015-01-26 20:00:06</tns:DeliveryDate>
        <tns:DeliveryMethod>5052d0b3-5ea3-46f2-823f-1472686a51dd</tns:DeliveryMethod>
        <tns:DeliveryTime>c 5 lj 10</tns:DeliveryTime>
        <tns:Department></tns:Department>
        <tns:Project></tns:Project>
        <tns:Recipient>
          <tns:Client>test</tns:Client>
          <tns:Official>тест</tns:Official>
          <tns:Address>
            <tns:Geography>cf862f56-442d-11dc-9497-0015170f8c09</tns:Geography>
            <tns:Info>тест</tns:Info>
            <tns:FreeForm>true</tns:FreeForm>
          </tns:Address>
          <tns:Phone>8-962-220-44-04</tns:Phone>
          <tns:Urgency>18c4f207-458b-11dc-9497-0015170f8c09</tns:Urgency>
          <tns:Cargo>
            <tns:CargoDescription>Осторожно хрупкий груз</tns:CargoDescription>
            <tns:CargoPackageQty>5</tns:CargoPackageQty>
            <tns:Weight>19.3</tns:Weight>
            <tns:VolumeWeight>18.5</tns:VolumeWeight>
            <tns:Length>5.0</tns:Length>
            <tns:Width>4.0</tns:Width>
            <tns:Height>1.2</tns:Height>
            </tns:Cargo>
            <tns:Products>
              <tns:Article>123-1</tns:Article>
              <tns:Name></tns:Name>
              <tns:Price>150</tns:Price>
              <tns:PackageQty>10</tns:PackageQty>
              <tns:Qty>5</tns:Qty>
            </tns:Products>
            <tns:Products>
              <tns:Article>123-3</tns:Article>
              <tns:Name></tns:Name>
              <tns:Price>5000</tns:Price>
              <tns:PackageQty>50</tns:PackageQty>
              <tns:Qty>5</tns:Qty>
            </tns:Products>
          </tns:Recipient>
        <tns:ReplyEMail>ordinary_user@mail.ru</tns:ReplyEMail>
        <tns:ReplySMSPhone>123</tns:ReplySMSPhone>
        <tns:Sender>
          <tns:Client>fgfgf</tns:Client>
          <tns:Official>fgfggg</tns:Official>
        </tns:Sender>
      </tns:OrderData>
    </tns:SaveWaybillOffice>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```




```
<tns:Address>
  <tns:Geography>aa4c67f8-442d-11dc-9497-0015170f8c09</tns:Geography>
  <tns:Info>ул. Советская д. 14, офис 222</tns:Info>
  <tns:FreeForm>>true</tns:FreeForm>
</tns:Address>
<tns:Phone>12-34-56</tns:Phone>
<tns:Info>superuser@gmail.com</tns:Info>
</tns:Sender>
<tns:TypeOfCargo>81dd8a13-8235-494f-84fd-9c04c51d50ec</tns:TypeOfCargo>
<tns:TypeOfPayer>1</tns:TypeOfPayer>
  <tns:WayOfPayment>1</tns:WayOfPayment>
  <tns:DeliveryOfCargo></tns:DeliveryOfCargo>
<tns:ParentOrderForWaybill></tns:ParentOrderForWaybill>
  <tns:ClientNumber></tns:ClientNumber>
  <tns:ArrayOfDocumentsPhoto>
    <tns:Document>паспорт</tns:Document>
    <tns:Required>1</tns:Required>
  </tns:ArrayOfDocumentsPhoto>
  <tns:ArrayOfDocumentsPhoto>
    <tns:Document>моряк</tns:Document>
    <tns:Required>0</tns:Required>
  </tns:ArrayOfDocumentsPhoto>
</tns:OrderData>
<tns:Office></tns:Office>
</tns:SaveWaybillOffice>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```



CALC: CALCULATION OF DELIVERY COST

The function calculates the delivery cost on passed parameters.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	data	data list for the calculation
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
ipaddress	string		Your address for the statistics

PARAMETER DESCRIPTION DATA

The structure «data» contains the data of one or several documents as a list of structures «Element». The structure «data» must have the property «Key» that contains the value «Documents». Parameters are put into a list using the property «List» that cannot be empty.

Every document from the «List» has the following properties:

Property	Type	Required	Description
Key	string	Yes	It always must be equal «Destination»
Fields	Element - list	Yes	Calculation parameters

List of order's properties in «Fields»:

Parameter (Key)	Value type	Required	Description
SenderGeography	string	Yes	GUID (Globally Unique Identifier) of the sender geography (how to get geographical objects see in the chapter GetReferenceData: geography) or of the client geography code if the client uses his own geography
RecipientGeography	string	Yes	GUID (Globally Unique Identifier) of the recipient geography (how to get geographical objects see in the chapter GetReferenceData: geography) or of the client geography code if the client uses his own geography
Urgency	string	No	GUID (Globally Unique Identifier) of the delivery's urgency (how to get a list of urgencies see in the chapter GetReferenceData: urgencies)
TypeOfCargo	string	Yes	GUID (Globally Unique Identifier) of cargo types (how to get a list of cargo types see in the chapter GetReferenceData: cargo types)
Weight	float	Yes	Cargo weight, kg.
Qty	float	Yes	Number of places, non-negative integer
deliverytype	string	No	Delivery type



ANSWER:

Element

«Element» has two properties: the property «Key» contains the value «Calc» and the property «List» contains information of the delivery cost. The property «Properties» contains the error information (see the chapter [Error information](#)).

Every answer of the list «List» contains the following properties:

Property	Type	Required	Description
Key	string	Yes	It always must be equal «Destination»
List	Element - list	Yes	Information of the found document or about an error (see the chapter Error information)

The tariff's information in the property «List» has the following values:

Parameter (Key)	Value type	Required	Description
Tariff	string		Element's GUID (Globally Unique Identifier) of the tariffs reference in the LMS «Cargo»
Total	string		Tariff total cost of the delivery
Urgency	string		Urgency GUID (Globally Unique Identifier) in the LMS «Cargo»
UrgencyName	string		Urgency name in the LMS «Cargo»
Currency	string		Tariff currency
MinPeriod	decimal		Minimum delivery period for the passed parameters
MaxPeriod	decimal		Maximum delivery period for the passed parameters
COD	boolean		True if this tariff allows to accept the cash on delivery
Return	boolean		True if this tariff allows to return the shipment
Agent	string		Agent name (if a third-party company delivers) that will deliver
Service	string		Name of the delivery service



EXAMPLE:

Request:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header/>
  <soap:Body>
    <m:Calc xmlns:m="http://www.cargo3.ru">
      <m:login>test</m:login>
      <m:password>2016</m:password>
      <m:data>
        <m:Key>Destinations</m:Key>
        <m>List>
          <m:Key>Destination</m:Key>
          <m:Fields>
            <m:Key>SenderGeography</m:Key>
            <m:Value>cf862f56-442d-11dc-9497-0015170f8c09</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Fields>
          <m:Fields>
            <m:Key>RecipientGeography</m:Key>
            <m:Value>cf862f56-442d-11dc-9497-0015170f8c09</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Fields>
          <m:Fields>
            <m:Key>TypeOfCargo</m:Key>
            <m:Value>4aab1fc6-fc2b-473a-8728-58bcd4ff79ba</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Fields>
          <m:Fields>
            <m:Key>Weight</m:Key>
            <m:Value>1</m:Value>
            <m:ValueType>float</m:ValueType>
          </m:Fields>
          <m:Fields>
            <m:Key>Qty</m:Key>
            <m:Value>1</m:Value>
            <m:ValueType>int</m:ValueType>
          </m:Fields>
        </m>List>
      </m:data>
      <m:parameters xmlns:xs="http://www.w3.org/2001/XMLSchema"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        <m:Key>Parameters</m:Key>
        <m>List>
          <m:Key>ipaddress</m:Key>
          <m:Value xsi:type="xs:string">10.0.0.1</m:Value>
          <m:ValueType>string</m:ValueType>
        </m>List>
      </m:parameters>
    </m:Calc>
  </soap:Body>
</soap:Envelope>
```



Answer:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Body>
    <m:CalcResponse xmlns:m="http://www.cargo3.ru">
      <m:return xmlns:xs="http://www.w3.org/2001/XMLSchema"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        <m:Key>Calc</m:Key>
        <m:List>
          <m:Key>Destination</m:Key>
          <m:Fields>
            <m:Key>SenderGeography</m:Key>
            <m:Value xsi:type="xs:string">cf862f56-442d-11dc-9497-
0015170f8c09</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Fields>
          <m:Key>RecipientGeography</m:Key>
            <m:Value xsi:type="xs:string">cf862f56-442d-11dc-9497-
0015170f8c09</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Fields>
          <m:Key>TypeOfCargo</m:Key>
            <m:Value xsi:type="xs:string">4aab1fc6-fc2b-473a-8728-
58bcd4ff79ba</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Fields>
          <m:Key>weight</m:Key>
            <m:Value xsi:type="xs:decimal">1</m:Value>
            <m:ValueType>float</m:ValueType>
          </m:Fields>
          <m:Key>Qty</m:Key>
            <m:Value xsi:type="xs:decimal">1</m:Value>
            <m:ValueType>float</m:ValueType>
          </m:Fields>
          <m:Key>Tariff</m:Key>
            <m:Value xsi:type="xs:string">6da21fe5-4f13-11dc-bda1-
0015170f8c09</m:Value>
            <m:Fields>
              <m:Key>Total</m:Key>
              <m:Value xsi:type="xs:decimal">350</m:Value>
              <m:ValueType>float</m:ValueType>
            </m:Fields>
          <m:Key>Currency</m:Key>
            <m:Value xsi:type="xs:string">ff3f7c38-4430-11dc-9497-
0015170f8c09</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Fields>
          <m:Key>Service</m:Key>
            <m:Value xsi:type="xs:string">Доставка по
Москва</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Fields>
          <m:Key>Urgency</m:Key>
            <m:Value xsi:type="xs:string">18c4f207-458b-11dc-9497-
0015170f8c09</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Fields>
          <m:Key>UrgencyName</m:Key>
            <m:Value xsi:type="xs:string">Срочная</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Fields>
          <m:Key>MinPeriod</m:Key>
            <m:Value xsi:type="xs:decimal">1</m:Value>
            <m:ValueType>float</m:ValueType>
          </m:Fields>
          <m:Key>MaxPeriod</m:Key>
            <m:Value xsi:type="xs:decimal">1</m:Value>
            <m:ValueType>float</m:ValueType>
          </m:Fields>
          <m:Key>COD</m:Key>
            <m:Value xsi:type="xs:boolean">true</m:Value>
            <m:ValueType>boolean</m:ValueType>
          </m:Fields>
          <m:Key>Return</m:Key>
```



```

        <m:Value xsi:type="xs:boolean">true</m:Value>
        <m:ValueType>boolean</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Agent</m:Key>
        <m:Value xsi:type="xs:string"/>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
</m>List>
<m>List>
    <m:Key>Tariff</m:Key>
    <m:Value xsi:type="xs:string">6da21fe6-4f13-11dc-bda1-
0015170f8c09</m:Value>
    <m:Fields>
        <m:Key>Total</m:Key>
        <m:Value xsi:type="xs:decimal">1250</m:Value>
        <m:ValueType>float</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Currency</m:Key>
        <m:Value xsi:type="xs:string">ff3f7c38-4430-11dc-9497-
0015170f8c09</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Service</m:Key>
        <m:Value xsi:type="xs:string">Автодоставка по
городу</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Urgency</m:Key>
        <m:Value xsi:type="xs:string">18c4f207-458b-11dc-9497-
0015170f8c09</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>UrgencyName</m:Key>
        <m:Value xsi:type="xs:string">Срочная</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>MinPeriod</m:Key>
        <m:Value xsi:type="xs:decimal">1</m:Value>
        <m:ValueType>float</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>MaxPeriod</m:Key>
        <m:Value xsi:type="xs:decimal">1</m:Value>
        <m:ValueType>float</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>COD</m:Key>
        <m:Value xsi:type="xs:boolean">true</m:Value>
        <m:ValueType>boolean</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Return</m:Key>
        <m:Value xsi:type="xs:boolean">true</m:Value>
        <m:ValueType>boolean</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Agent</m:Key>
        <m:Value xsi:type="xs:string"/>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
</m>List>
<m>List>
    <m:Key>Tariff</m:Key>
    <m:Value xsi:type="xs:string">6da21fe6-4f13-11dc-bda1-
0015170f8c09</m:Value>
    <m:Fields>
        <m:Key>Total</m:Key>
        <m:Value xsi:type="xs:decimal">1250</m:Value>
        <m:ValueType>float</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Currency</m:Key>
        <m:Value xsi:type="xs:string">ff3f7c38-4430-11dc-9497-
0015170f8c09</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Service</m:Key>
        <m:Value xsi:type="xs:string">Автодоставка по
городу</m:Value>
        <m:ValueType>string</m:ValueType>

```



```

0015170f8c09</m:Value>
</m:Fields>
<m:Fields>
  <m:Key>Urgency</m:Key>
  <m:Value xsi:type="xs:string">9d7b9bdc-772f-11dc-a1ad-
    <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>UrgencyName</m:Key>
  <m:Value xsi:type="xs:string">Суточная</m:Value>
  <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>MinPeriod</m:Key>
  <m:Value xsi:type="xs:decimal">1</m:Value>
  <m:ValueType>float</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>MaxPeriod</m:Key>
  <m:Value xsi:type="xs:decimal">1</m:Value>
  <m:ValueType>float</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>COD</m:Key>
  <m:Value xsi:type="xs:boolean">true</m:Value>
  <m:ValueType>boolean</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>Return</m:Key>
  <m:Value xsi:type="xs:boolean">true</m:Value>
  <m:ValueType>boolean</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>Agent</m:Key>
  <m:Value xsi:type="xs:string"/>
  <m:ValueType>string</m:ValueType>
</m:Fields>
</m:List>
</m:List>
</m:return>
</m:CalcResponse>
</soap:Body>
</soap:Envelope>

```



GETFORMSFORDOCUMENTS: WAYBILL SCAN

The function gets the waybill scan.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	documents	document list
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
DocumentType	string		Type of tracked documents can be «Order»/«Orders» or «Waybill»/«Waybills»
Type	string	Yes	It must have the value «scan»
Format	string		Format of the returned document (PDF by default)
NumberType	string		Number type is used for the document search. If the number type is empty then the search will be carried out by the internal number. Available values: «ClientNumber» (client number), «AgentNumber» (agent number), «OrderNumber» (order number, it is used only for the waybill's search) and «PackageID» (id of the package)
OnlySelectedType	boolean		True if the search is carried out strictly by the number of NumberType . If it is false and the function cannot find any document by the number of NumberType then the search will be carried out by the internal number from the LMS «Cargo».

PARAMETER DESCRIPTION DOCUMENTS

The structure «documents» contains the data of one or several documents as a list of structures «Element». The structure «documents» must have the property «Key» that contains the value «Documents». Parameters are put into a list using the property «List» that cannot be empty.

Every document from the «List» has the following properties:

Property	Type	Required	Description
Key	string	Yes	Document number. The function returns the document's scan by this number.
Properties	Element - list	No	Passed parameters that will be used only for this document number (parameters set is the same as in total parameters). It is used to pass several documents of different types or with different number types (the client number or the internal number from the LMS «Cargo»)



ANSWER:

Element

«Element» has two properties: the property «Key» contains the value «GetPrintForms» and the property «List» contains information of the found document and the scan. The property «Properties» contains the error information (see the chapter [Error information](#)).

Every answer of the list «List» has the following properties:

Property	Type	Required	Description
Key	string	Yes	It always must be equal «GetPrintForms»
List	Element - list	Yes	Information of the found document or about an error (see the chapter Error information)

The information of the found document in the property «List» has the following values:

Parameter (Key)	Value type	Required	Description
Key	string	Yes	Element's GUID (Globally Unique Identifier) of the tariffs reference in the LMS «Cargo»
Value	string	Yes	Tariff total cost of the delivery
Properties	Element - list	Yes	Information of the found document or about an error (see the chapter Error information)

The information of the found document in the property «Properties» has the following values:

Parameter (Key)	Value type	Required	Description
DocumentType	string		Type of the passed document
DocumentName	string		Internal document name in the LMS «Cargo»
GUID	string		GUID (Globally Unique Identifier) of the document in the LMS «Cargo»
Date	binardata		Document data in the LMS «Cargo»
Number	string		Internal document number in the LMS «Cargo»
FormName	string		Name of the print form
FormFormat	string		Format of the returned scan
BData	binardata		Document scan in binary format



EXAMPLE:

Request:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header/>
  <soap:Body>
    <m:GetFormsForDocuments xmlns:m="http://www.cargo3.ru">
      <m:login>Login</m:login>
      <m:password>password</m:password>
      <m:documents>
        <m:Key>Documents</m:Key>
        <m:List>
          <m:Key>495-BB-A4422219314</m:Key>
        </m:List>
      </m:documents>
      <m:parameters>
        <m:Key>Parameters</m:Key>
        <m:List>
          <m:Key>DocumentType</m:Key>
          <m:Value>waybill</m:Value>
          <m:ValueType>string</m:ValueType>
        </m:List>
        <m:List>
          <m:Key>Type</m:Key>
          <m:Value>scan</m:Value>
          <m:ValueType>string</m:ValueType>
        </m:List>
        <m:List>
          <m:Key>OnlyPath</m:Key>
          <m:Value>>false</m:Value>
          <m:ValueType>boolean</m:ValueType>
        </m:List>
        <m:List>
          <m:Key>Format</m:Key>
          <m:Value>PDF</m:Value>
          <m:ValueType>string</m:ValueType>
        </m:List>
      </m:parameters>
    </m:GetFormsForDocuments>
  </soap:Body>
</soap:Envelope>
```

Answer:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Body>
    <m:GetFormsForDocumentsResponse xmlns:m="http://www.cargo3.ru">
      <m:return xmlns:xs="http://www.w3.org/2001/XMLSchema"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        <m:Key>GetPrintForms</m:Key>
        <m:List>
          <m:Key>495-BB-A4422219314</m:Key>
          <m:Value xsi:type="xs:string">Waybill</m:Value>
          <m:ValueType>string</m:ValueType>
          <m:Properties>
            <m:Key>DocumentType</m:Key>
            <m:Value xsi:type="xs:string">waybill</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Properties>
          <m:Properties>
            <m:Key>DocumentName</m:Key>
            <m:Value xsi:type="xs:string">накладная</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Properties>
          <m:Properties>
            <m:Key>GUID</m:Key>
            <m:Value xsi:type="xs:string">6b28f212-4fad-11e4-9c33-
005056b1a128</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Properties>
          <m:Properties>
            <m:Key>Date</m:Key>
            <m:Value xsi:type="xs:dateTime">2014-10-09T16:11:38</m:Value>
            <m:ValueType>dateTime</m:ValueType>
          </m:Properties>
          <m:Properties>
            <m:Key>Number</m:Key>
            <m:Value xsi:type="xs:string">495-BB-A4422219314 </m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Properties>
          <m:Properties>
            <m:Key>FormName</m:Key>
          </m:Properties>
        </m:List>
      </m:return>
    </m:GetFormsForDocumentsResponse>
  </soap:Body>
</soap:Envelope>
```



```
        <m:Key>FormFormat</m:Key>
        <m:Value xsi:type="xs:string">PDF</m:Value>
        <m:ValueType>string</m:ValueType>
    </m:Properties>
<m:BData>255044462D312E350A25E2.....7274787265660A39313332340A2525454F46</m:BData>
    </m:List>
  </m:return>
</m:GetFormsForDocumentsResponse>
</soap:Body>
</soap:Envelope>
```



GETDOCUMENTS: DOCUMENT INFORMATION

The function gets the document information.

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	data	documents list
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

Parameters:

Parameter (Key)	Value type	Required	Description
DocumentType	string	Yes	Type of the tracked documents can be «Order»/«Orders» or «Waybill»/«Waybills»
NumberType	string		Number type is used for the document search. If the number type is empty then the search will be carried out by the internal number. Available values: «ClientNumber» (client number), «AgentNumber» (agent number), «OrderNumber» (order number, it is used only for the waybill's search) and «PackageID» (id of the package)
OnlySelectedType	boolean		True if the search is carried out strictly by the number of NumberType . If it is false and the function cannot find any document by the number of NumberType then the search will be carried out by the internal number from the LMS «Cargo».

PARAMETER DESCRIPTION DATA

The structure «data» contains the data of one or several documents as a list of structures «Element». The structure «data» must have the property «Key» that contains the value «Number». Parameters are put into a list using the property «List» that cannot be empty.

Every document from the «List» has the following properties:

Property	Type	Required	Description
Number	string	Yes	Document number. The function returns the document's information by this number.
Properties	Element - list	Yes	Document number. The function returns the document's information by this number.

ANSWER:

Element

«Element» has two properties: the property «Key» contains the value «GetDocuments» and the property «Tables» contains information of the found document. The property «Properties» contains the error information (see the chapter [Error information](#)).



EXAMPLE:

Request:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header/>
  <soap:Body>
    <m:GetDocuments xmlns:m="http://www.cargo3.ru">
      <m:login>СИТИБАНК 2013</m:login>
      <m:password>0QqEZPdrNaGalBy</m:password>
      <m:data>
        <m:Key>Number</m:Key>
        <m>List>
          <m:Key>888-39879-0000000442</m:Key>
          <m:Properties>
            <m:Key>Number</m:Key>
            <m:Value>888-39879-0000000442</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Properties>
        </m>List>
      </m:data>
      <m:parameters>
        <m:Key>Parameters</m:Key>
        <m>List>
          <m:Key>DocumentType</m:Key>
          <m:Value>waybill</m:Value>
          <m:ValueType>string</m:ValueType>
        </m>List>
      </m:parameters>
    </m:GetDocuments>
  </soap:Body>
</soap:Envelope>
```

Answer:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Body>
    <m:GetDocumentsResponse xmlns:m="http://www.cargo3.ru">
      <m:return xmlns:xs="http://www.w3.org/2001/XMLSchema"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        <m:Key>GetDocuments</m:Key>
        <m:Tables>
          <m:Key>waybill</m:Key>
          <m:Properties>
            <m:Key>DocumentType</m:Key>
            <m:Value xsi:type="xs:string">waybill</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Properties>
          <m:Properties>
            <m:Key>Number</m:Key>
            <m:Value xsi:type="xs:string">888-39879-0000000442</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Properties>
          <m:Properties>
            <m:Key>DocumentName</m:Key>
            <m:Value xsi:type="xs:string">Накладная</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Properties>
          <m:Properties>
            <m:Key>GUID</m:Key>
            <m:Value xsi:type="xs:string">8b16a241-415f-11e6-8656-
005056b1a128</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Properties>
          <m:Properties>
            <m:Key>Date</m:Key>
            <m:Value xsi:type="xs:dateTime">2016-07-03T23:48:56</m:Value>
            <m:ValueType>dateTime</m:ValueType>
          </m:Properties>
          <m:Properties>
            <m:Key>Number</m:Key>
            <m:Value xsi:type="xs:string">888-39879-0000000442</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Properties>
          <m:Properties>
            <m:Key>DataVersion</m:Key>
            <m:Value xsi:type="xs:string">AAAAAACyqPQ=</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Properties>
          <m:Properties>
            <m:Key>DeletionMark</m:Key>
            <m:Value xsi:type="xs:boolean">>false</m:Value>
            <m:ValueType>boolean</m:ValueType>
          </m:Properties>
        </m:Tables>
      </m:return>
    </m:GetDocumentsResponse>
  </soap:Body>
</soap:Envelope>
```



```

</m:Properties>
<m:Properties>
  <m:Key>Posted</m:Key>
  <m:Value xsi:type="xs:boolean">true</m:Value>
  <m:ValueType>boolean</m:ValueType>
</m:Properties>
<m:Properties>
  <m:Key>AgentInfoURL</m:Key>
  <m:Value xsi:type="xs:string"/>
  <m:ValueType>string</m:ValueType>
</m:Properties>
<m:Properties>
  <m:Key>HasScan</m:Key>
  <m:Value xsi:type="xs:boolean">>false</m:Value>
  <m:ValueType>boolean</m:ValueType>
</m:Properties>
<m:Fields>
  <m:Key>Office</m:Key>
  <m:Value xsi:type="xs:string">ba19c6c4-44c4-11dc-9497-
0015170f8c09</m:Value>
  <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>Department</m:Key>
  <m:Value xsi:type="xs:string">80009-ук-дз</m:Value>
  <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>TakeDate</m:Key>
  <m:Value xsi:type="xs:dateTime">2016-08-29T00:00:00</m:Value>
  <m:ValueType>dateTime</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>Comment</m:Key>
  <m:Value xsi:type="xs:string">Возврат груза: 14</m:Value>
  <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>Sender</m:Key>
  <m:Value xsi:type="xs:string">000 "Торговая компания
ИНАГРО" </m:Value>
  <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>SenderOfficial</m:Key>
  <m:Value xsi:type="xs:string">Кузин Дмитрий</m:Value>
  <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>SenderGeography</m:Key>
  <m:Value xsi:type="xs:string">84055812-4429-11dc-9497-
0015170f8c09</m:Value>
  <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>SenderAddress</m:Key>
  <m:Value xsi:type="xs:string">ул А Германа 15</m:Value>
  <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>SenderPhone</m:Key>
  <m:Value xsi:type="xs:string">+7 (812) 318-36-26 (доб.
306) </m:Value>
  <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>SenderEMail</m:Key>
  <m:Value xsi:type="xs:string"/>
  <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>SenderInfo</m:Key>
  <m:Value xsi:type="xs:string"/>
  <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>Recipient</m:Key>
  <m:Value xsi:type="xs:string">Ситибанк</m:Value>
  <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>RecipientOfficial</m:Key>
  <m:Value xsi:type="xs:string">Егорова А.В.</m:Value>
  <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>RecipientGeography</m:Key>

```



```

0015170f8c09</m:Value>
    <m:Value xsi:type="xs:string">cf862f56-442d-11dc-9497-
    <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
    <m:Key>RecipientAddress</m:Key>
    <m:Value xsi:type="xs:string">Средиземноморский
проект!!!!!!!!!!!!!!!!!!!!!!</m:Value>
    <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
    <m:Key>RecipientPhone</m:Key>
    <m:Value xsi:type="xs:string">0202020202</m:Value>
    <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
    <m:Key>RecipientEMail</m:Key>
    <m:Value xsi:type="xs:string"/>
    <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
    <m:Key>RecipientInfo</m:Key>
    <m:Value xsi:type="xs:string"/>
    <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
    <m:Key>Urgency</m:Key>
    <m:Value xsi:type="xs:string">18c4f207-458b-11dc-9497-
0015170f8c09</m:Value>
    <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
    <m:Key>Payer</m:Key>
    <m:Value xsi:type="xs:string">0</m:Value>
    <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
    <m:Key>PaymentMethod</m:Key>
    <m:Value xsi:type="xs:string">1</m:Value>
    <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
    <m:Key>ShippingMethod</m:Key>
    <m:Value xsi:type="xs:string">e45b6d73-fd62-44da-82a6-
44eb4d1d9490</m:Value>
    <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
    <m:Key>TypeOfCargo</m:Key>
    <m:Value xsi:type="xs:string">4aab1fc6-fc2b-473a-8728-
58bcd4ff79ba</m:Value>
    <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
    <m:Key>WithReturn</m:Key>
    <m:Value xsi:type="xs:boolean">>false</m:Value>
    <m:ValueType>boolean</m:ValueType>
</m:Fields>
<m:Fields>
    <m:Key>Weight</m:Key>
    <m:Value xsi:type="xs:decimal">0.1</m:Value>
    <m:ValueType>float</m:ValueType>
</m:Fields>
<m:Fields>
    <m:Key>Height</m:Key>
    <m:Value xsi:type="xs:decimal">0</m:Value>
    <m:ValueType>float</m:ValueType>
</m:Fields>
<m:Fields>
    <m:Key>Length</m:Key>
    <m:Value xsi:type="xs:decimal">0</m:Value>
    <m:ValueType>float</m:ValueType>
</m:Fields>
<m:Fields>
    <m:Key>Width</m:Key>
    <m:Value xsi:type="xs:decimal">0</m:Value>
    <m:ValueType>float</m:ValueType>
</m:Fields>
<m:Fields>
    <m:Key>CargoDescription</m:Key>
    <m:Value xsi:type="xs:string"/>
    <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
    <m:Key>CargoPackageQty</m:Key>
    <m:Value xsi:type="xs:decimal">2</m:Value>
    <m:ValueType>float</m:ValueType>

```



```

</m:Fields>
<m:Fields>
  <m:Key>InsuranceRate</m:Key>
  <m:Value xsi:type="xs:decimal">10000</m:Value>
  <m:ValueType>float</m:ValueType>
</m:Fields>
0015170f8c09</m:Value>
  <m:Key>InsuranceRateCurrency</m:Key>
  <m:Value xsi:type="xs:string">ff3f7c38-4430-11dc-9497-
    <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>DeclaredValueRate</m:Key>
  <m:Value xsi:type="xs:decimal">0</m:Value>
  <m:ValueType>float</m:ValueType>
</m:Fields>
0015170f8c09</m:Value>
  <m:Key>DeclaredValueRateCurrency</m:Key>
  <m:Value xsi:type="xs:string">ff3f7c38-4430-11dc-9497-
    <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>ValueForCustomsPurposes</m:Key>
  <m:Value xsi:type="xs:decimal">0</m:Value>
  <m:ValueType>float</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>COD</m:Key>
  <m:Value xsi:type="xs:decimal">0</m:Value>
  <m:ValueType>float</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>ReplyEMail</m:Key>
  <m:Value xsi:type="xs:string"/>
  <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Fields>
  <m:Key>ReplySMSPhone</m:Key>
  <m:Value xsi:type="xs:string"/>
  <m:ValueType>string</m:ValueType>
</m:Fields>
<m:Tables>
  <m:Key>CargoPackages</m:Key>
  <m>List>
    <m:Key>CargoPackage</m:Key>
    <m:Fields>
      <m:Key>PackageID</m:Key>
      <m:Value xsi:type="xs:string">1234545</m:Value>
      <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
      <m:Key>Length</m:Key>
      <m:Value xsi:type="xs:decimal">45</m:Value>
      <m:ValueType>float</m:ValueType>
    </m:Fields>
    <m:Fields>
      <m:Key>width</m:Key>
      <m:Value xsi:type="xs:decimal">34</m:Value>
      <m:ValueType>float</m:ValueType>
    </m:Fields>
    <m:Fields>
      <m:Key>Height</m:Key>
      <m:Value xsi:type="xs:decimal">245</m:Value>
      <m:ValueType>float</m:ValueType>
    </m:Fields>
    <m:Fields>
      <m:Key>weight</m:Key>
      <m:Value xsi:type="xs:decimal">10</m:Value>
      <m:ValueType>float</m:ValueType>
    </m:Fields>
    <m:Fields>
      <m:Key>PackageQty</m:Key>
      <m:Value xsi:type="xs:decimal">1</m:Value>
      <m:ValueType>float</m:ValueType>
    </m:Fields>
  </m>List>
  <m>List>
    <m:Key>CargoPackage</m:Key>
    <m:Fields>
      <m:Key>PackageID</m:Key>
      <m:Value xsi:type="xs:string">1234546</m:Value>
      <m:ValueType>string</m:ValueType>
    </m:Fields>
    <m:Fields>
      <m:Key>Length</m:Key>

```




```

        <m:Value xsi:type="xs:decimal">67</m:Value>
        <m:ValueType>float</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>width</m:Key>
        <m:Value xsi:type="xs:decimal">56</m:Value>
        <m:ValueType>float</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Height</m:Key>
        <m:Value xsi:type="xs:decimal">35</m:Value>
        <m:ValueType>float</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>Weight</m:Key>
        <m:Value xsi:type="xs:decimal">5</m:Value>
        <m:ValueType>float</m:ValueType>
    </m:Fields>
    <m:Fields>
        <m:Key>PackageQty</m:Key>
        <m:Value xsi:type="xs:decimal">1</m:Value>
        <m:ValueType>float</m:ValueType>
    </m:Fields>
</m>List>
</m:Tables>
<m:Tables>
    <m:Key>Services</m:Key>
    <m>List>
        <m:Key>Service</m:Key>
        <m:Fields>
            <m:Key>ServiceID</m:Key>
            <m:Value xsi:type="xs:string">6da21fe8-4f13-11dc-
bda1-0015170f8c09</m:Value>
            <m:ValueType>string</m:ValueType>
        </m:Fields>
        <m:Fields>
            <m:Key>ServiceName</m:Key>
            <m:Value xsi:type="xs:string">доставка по
России</m:Value>
            <m:ValueType>string</m:ValueType>
        </m:Fields>
        <m:Fields>
            <m:Key>Total</m:Key>
            <m:Value xsi:type="xs:decimal">13972</m:Value>
            <m:ValueType>float</m:ValueType>
        </m:Fields>
    </m>List>
    <m>List>
        <m:Key>Service</m:Key>
        <m:Fields>
            <m:Key>ServiceID</m:Key>
            <m:Value xsi:type="xs:string">cc03d9ad-61f4-11dc-
bda1-0015170f8c09</m:Value>
            <m:ValueType>string</m:ValueType>
        </m:Fields>
        <m:Fields>
            <m:Key>ServiceName</m:Key>
            <m:Value xsi:type="xs:string">Страхование
груза</m:Value>
            <m:ValueType>string</m:ValueType>
        </m:Fields>
        <m:Fields>
            <m:Key>Total</m:Key>
            <m:Value xsi:type="xs:decimal">50</m:Value>
            <m:ValueType>float</m:ValueType>
        </m:Fields>
    </m>List>
    <m>List>
        <m:Key>Service</m:Key>
        <m:Fields>
            <m:Key>ServiceID</m:Key>
            <m:Value xsi:type="xs:string">7ad2661d-bfcd-11df-
9948-00237dd28494</m:Value>
            <m:ValueType>string</m:ValueType>
        </m:Fields>
        <m:Fields>
            <m:Key>ServiceName</m:Key>
            <m:Value xsi:type="xs:string">Возврат
документов</m:Value>
            <m:ValueType>string</m:ValueType>
        </m:Fields>
        <m:Fields>
            <m:Key>Total</m:Key>
            <m:Value xsi:type="xs:decimal">150</m:Value>
            <m:ValueType>float</m:ValueType>
        </m:Fields>
    </m>List>

```



```
</m:Tables>
  </m:Tables>
</m:return>
</m:GetDocumentsResponse>
</soap:Body>
</soap:Envelope>
```



UPDATECLIENTPRODUCTS: CREATING NEW OR UPDATING EXISTING GOODS

The function allows to create the new counterparty's goods or update some information already stored goods. The method works as follows: if the function cannot find any goods by the passed article then it creates the new goods. Otherwise it updates data of the found item. To update goods information you can pass the article or GUID (Globally Unique Identifier) from the LMS «Cargo».

REQUEST PARAMETERS:

string	login	user name
string	password	password
Element	data	goods list
Element	parameters	parameters list

PARAMETER DESCRIPTION PARAMETERS

The structure «parameters» must have the property «Key» that contains the value «parameters». Parameters are put into a list using the property «List» that cannot be empty.

PARAMETER DESCRIPTION DATA

The structure «data» contains the data of one or several documents as a list of structures «Element». The structure «data» must have the property «Key» that contains the value «Products». Parameters are put into a list using the property «List» that cannot be empty.

Every document from the «List» has the following properties:

Property	Type	Required	Description
Key	string	Yes	Goods article that you wish to create or update.
Fields	Element - list	Yes	Passed parameters of goods

Order's properties passed through «Fields»:

Parameter (Key)	Value type	Required	Description
Article	string	Yes	Goods article that you wish to create or update. This value can be changed only in the LMS «Cargo». Maximum is 50 symbols.
Name	string	Yes	Goods name. This field can be changed while updating goods information. Maximum is 150 symbols.
FullName	string	Yes	Goods full name. This field can be changed while updating goods information.
BaseUnit	string	Yes	Basic measurement unit of goods. You can put the code of the measurement unit from the IS' classifier or GUID (Globally Unique Identifier) of the measurement unit from the LMS «Cargo». This value can be changed only in the LMS «Cargo».
Project	string	No	GUID of the client project. It is used in cases of keep goods account in the context of client projects. This value can be changed only in the LMS «Cargo».
Description	string	No	Goods additional description. This field can be changed while updating goods information.
Party	boolean	No	True if the batch accounting is uses for goods. This value can be changed only in the LMS «Cargo».



ANSWER:

Element

«Element» has two properties: the property «Key» contains the value «UpdateClientProducts» and the property «List» contains information of the found document and the scan. The property «Properties» contains the error information (see the chapter [Error information](#)).

Every answer of the list «List» has the following properties:

Property	Type	Required	Description
Key	string	Yes	It always must be equal «GetPrintForms»
List	Element - list	Yes	Information of the found document or about an error (see the chapter Error information)

The information of the found document in the property «List» has the following values:

Parameter (Key)	Value type	Required	Description
Key	string	Yes	Element's GUID (Globally Unique Identifier) of the tariffs reference in the LMS «Cargo»
Value	string	Yes	Tariff total cost for delivery
Properties	Element - list	Yes	Information of the found document or about an error (see the chapter Error information)

The information of the found document in the property «Properties» has the following values:

Parameter (Key)	Value type	Required	Description
Article	string		Goods article
Type	string		Processing method of goods (NEW, UPDATE)
GUID	string		GUID (Globally Unique Identifier) in the LMS «Cargo»
BaseUnit	List		Information about the goods measurement unit in the LMS «Cargo»



EXAMPLE:

Request:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header/>
  <soap:Body>
    <m:UpdateClientProducts xmlns:m="http://www.cargo3.ru">
      <m:Login>test</m:Login>
      <m>Password>2016</m>Password>
      <m:data>
        <m:Key>Productss</m:Key>
        <m>List>
          <m:Key>M053115302w22</m:Key>
          <m:Fields>
            <m:Key>Article</m:Key>
            <m:Value>M053115302w22</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Fields>
          <m:Fields>
            <m:Key>Name</m:Key>
            <m:Value>test</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Fields>
          <m:Fields>
            <m:Key>FullName</m:Key>
            <m:Value>test</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Fields>
          <m:Fields>
            <m:Key>BaseUnit</m:Key>
            <m:Value>796</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Fields>
        </m>List>
        <m>List>
          <m:Key>M053115302w21</m:Key>
          <m:Fields>
            <m:Key>Article</m:Key>
            <m:Value>M053115302w21</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Fields>
          <m:Fields>
            <m:Key>Name</m:Key>
            <m:Value>tes2t</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Fields>
          <m:Fields>
            <m:Key>FullName</m:Key>
            <m:Value>test33</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Fields>
        </m>List>
      </m:data>
      <m:parameters>
        <m:Key>Parameters</m:Key>
      </m:parameters>
    </m:UpdateClientProducts>
  </soap:Body>
</soap:Envelope>
```

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Body>
    <m:UpdateClientProductsResponse xmlns:m="http://www.cargo3.ru">
      <m:return xmlns:xs="http://www.w3.org/2001/XMLSchema"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        <m:Key>UpdateClientProducts</m:Key>
        <m>List>
          <m:Key>Product</m:Key>
          <m:Properties>
            <m:Key>Article</m:Key>
            <m:Value xsi:type="xs:string">M053115302w22</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Properties>
          <m:Properties>
            <m:Key>Type</m:Key>
            <m:Value xsi:type="xs:string">NEW</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Properties>
          <m:Properties>
            <m:Key>GUID</m:Key>
            <m:Value xsi:type="xs:string">e0f9bcf4-897f-11e6-a9b9-
005056b1a128</m:Value>
            <m:ValueType>string</m:ValueType>
          </m:Properties>
        </m>List>
      </m:return>
    </m:UpdateClientProductsResponse>
  </soap:Body>
</soap:Envelope>
```



```

    </m:Properties>
    <m>List>
      <m:Key>BaseUnit</m:Key>
      <m:Properties>
        <m:Key>Name</m:Key>
        <m:Value xsi:type="xs:string">шт</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Properties>
      <m:Properties>
        <m:Key>GUID</m:Key>
        <m:Value xsi:type="xs:string">e0f9bcf5-897f-11e6-a9b9-
005056b1a128</m:Value>
          <m:ValueType>string</m:ValueType>
        </m:Properties>
      </m>List>
    </m>List>
    <m>List>
      <m:Key>Product</m:Key>
      <m:Properties>
        <m:Key>Article</m:Key>
        <m:Value xsi:type="xs:string">M053115302w21</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Properties>
      <m:Properties>
        <m:Key>Type</m:Key>
        <m:Value xsi:type="xs:string">Update</m:Value>
        <m:ValueType>string</m:ValueType>
      </m:Properties>
      <m:Properties>
        <m:Key>GUID</m:Key>
        <m:Value xsi:type="xs:string">b8d2a7a4-897f-11e6-a9b9-
005056b1a128</m:Value>
          <m:ValueType>string</m:ValueType>
        </m:Properties>
      </m>List>
    </m:return>
  </m:UpdateClientProductsResponse>
</soap:Body>
</soap:Envelope>

```

