

Contents

- 1 Introduction
 - 1.1 Purpose
 - 1.2 Requirements
- 2 Protocol
 - 2.1 XML-RPC
 - 2.2 General Protocol Provisions
 - 2.3 Authentication
 - 2.4 Restrictions
 - 2.5 Supported Methods
- 3 Methods
 - 3.1.1 areaCodes.listRegions
 - 3.2.1 customers.listCustomers
 - 3.2.2 customers.addCustomer
 - 3.2.3 customers.updateCustomer
 - 3.2.4 customers.addPayment
 - 3.2.5 customers.deleteCustomer
 - 3.3.1 equipment.listGateways
 - 3.3.2 equipment.addSIPEquipment
 - 3.3.3 equipment.addH323Equipment
 - 3.3.4 equipment.updateSIPEquipment
 - 3.3.5 equipment.updateH323Equipment
 - 3.3.6 equipment.deleteEquipment
 - 3.4.1 routing.listRoutes
 - 3.4.2 routing.addRoutes
 - 3.4.3 routing.updateRoutes
 - 3.4.4 routing.deleteRoutes
 - 3.5.1 cdr.listSuccessCalls
 - 3.5.2 cdr.listFailedCalls
 - 3.5.3 cdr.listAggregate
 - 3.5.4 cdr.listSummary

Appendixes

- Appendix A - Release Notes
- Appendix B - Requests Examples

1 Introduction

1.1 Purpose

Hostedswitch API is a set of predefined methods and an environment to call these methods used to facilitate usage of the Hostedswitch routing/billing system by means other than Hostedswitch user-friendly web interface, such as member's proprietary accounting systems, web robots, etc.

This specification defines the protocol for communication with API server and describes the methods provided by the API version 1 (later referred to as APIv1).

1.2 Requirements

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

2 Protocol

2.1 XML-RPC

Hostedswitch API Communication Protocol is based on well-known XML-RPC:

<http://www.xmlrpc.com/spec>

which was chosen for its ability to work seamlessly over HTTP, simplicity and clarity (vs SOAP or similar protocols).

2.2 General Protocol Provisions

An XML-RPC message is an HTTP-POST request. The body of the request is in XML. A procedure executes on the server and the value it returns is also formatted in XML.

The request to APIv1 should be sent to the following HTTP URL:

```
API-URL = https://hostedswitch.com/hs/APIv1
```

There is also the so called "nowrite" URL

```
API-URL-nowrite = https://hostedswitch.com/hs/APIv1?nowrite=1
```

which differs from API-URL in that any write method called via this URL will not actually save any changes it is intended to make. This is useful for testing and debugging purposes.

The request MUST be sent over SSL encrypted connection and the client SHOULD verify the validity of the server's certificate to ensure security of the transmission.

The POST request MUST contain correct Host (=hostedswitch.com), Content-Type (=text/xml) and Content-Length headers

```
POST /hs/APIv1 HTTP/1.0
Host: hostedswitch.com
Content-Type: text/xml
Content-length: 123
```

The encoding for XML content is not constrained, although usage of encodings other than ISO-8859-1 or symbols incompatible with ISO-8859-1 is discouraged as the server will attempt to convert strings to ISO-8859-1 and the data are likely to be corrupted.

The body of the request MUST contain well-formed XML, it is recommended that this XML contains no formatting (i.e. no insignificant white-spaces) for the sake of compactness, since requests may often be bulky and may be rejected by the HTTP server due to their size. General form of the request data is

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<methodCall>
  <methodName>method-name</methodName>
  <params>
    <param>
      <value>
        <struct>
          <member><name>request-parameter-1</name><value><int>1</int></value></member>
          ...
        </struct>
      </value>
    </param>
  </params>
</methodCall>
```

where method-name is the name of the method to call (methods are described later).

Each method accepts a single parameter which is a STRUCT (the call will fail if <params> container in the above request holds more than one <param>, or if the first <param> is not a STRUCT). Later in the document we will refer to members of that <struct> as request parameters.

The response to the correct request is a <HTTP 200 OK> reply with data containing the response body unless an unrecoverable internal error occurs, in which case empty <HTTP 500 Internal Server Error> reply is sent.

The response body is a valid XML in ISO-8859-1 encoding. Unlike the example below response XML will not contain any formatting (i.e. no insignificant white-spaces at all) for the sake of compactness, since responses may often be bulky. Response contains a single <params> container which holds single <param> which value in turn is a STRUCT

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<methodResponse>
  <params>
    <param>
      <value>
        <struct>
```

```

    <member><name>response-parameter-1</name><value><int>1</int></value></member>
    ...
  </struct>
</value>
</param>
</params>
</methodResponse>

```

Members of that STRUCT are referred later in this document as response parameters.

In case of recoverable fault during request processing an XML-RPC fault response is sent with "faultString" parameter describing the error and usually giving a hint at where it occurred.

We will specify required (of preferred) types of request parameters and describe types of response parameters in the methods' specification sections. Whenever the type is not explicitly specified, it is assumed to be STRING. Client implementors SHOULD follow the types recommendations closely, failing to do so may lead to unexpected and hard-to-debug conversion errors.

The API server is rather strict to parameter validation and error recovery. If some of the request parameters are found to be invalid, the fault response is sent immediately without attempt to process the rest of the request (although unrecognized input parameters are silently ignored; client implementors are discouraged to send excess parameters as it may potentially lead to unexpected behavior and increases the request's size). If an error occurs while processing the input data (for example, while route is being saved) the fault response is sent and all the changes made so far by the request are rolled back.

2.3 Authentication

In this version of API authentication is done in each request separately. The following two simple methods of authentication are supported

```

Plain
MD5

```

No matter which one is used for each request, the following 3 parameters MUST be included (please note that the names start with a dot)

```

.authMethod - INTEGER REQUIRED
  It is equal to 1 for Plain authentication method or 2 for MD5.

.login      - STRING REQUIRED
  That is the desired login of the Hostedswitch Role under which API request will
  be fulfilled. Both Agreement Supervisor login (ex: "HS12345") and employee
  login (ex: "HS12345joe") are accepted but in case of Employee login the Employee
  Role should have suitable rights to make the request and the Password for the Role
  should be specified under Rights & Roles section of the Hostedswitch web interface.

.password   - STRING REQUIRED
  For Plain Authentication method this must be the password corresponding to ".login".
  For MD5 method this must be the auth-hash (two or more strings separated by space
  represent the concatenation of those strings)

  auth-hash := HEX(MD5( .login ":" salt ":" password))
  salt := "HOSTEDSWITCH"

```

Here MD5 is a MD5 hash function with binary result, HEX is a function converting binary string to hexadecimal representation (no matter, uppercase or lowercase). Password is the password corresponding to ".login".

The MD5 method is more secure as it avoids storing and sending open passwords. Given the ".login" = "HS12345joe", password for that role = "secret" and MD5 Authentication method

```

.password = HEX(MD5("HS12345joe:HOSTEDSWITCH:secret")) = "7EB7032F292CE413862BD85B44144703"

```

2.4 Restrictions

The API server will only serve one request at a time per account. Attempting to send another simultaneous request will result in a "There is a request in progress already" fault.

The API server will track the rate of the incoming requests, separately for each requested method, so as to control system load and to avoid denial of service. The rate is determined by calculating running average for rate of 50 consecutive requests. When it exceeds warning limit of approximately

3 requests/minute for most of the methods, special parameter is put into the response (note that the name starts with a dot)

`.requestRateWarning` - INTEGER

to alert client of dangerously high request rate. The value of the parameter is the recommended minimal delay (in seconds) between consecutive requests to the method. If the rate exceeds hardstop limit of approximately 5 requests/minute, the request will result in "Requests too frequent" fault. Such scheme effectively protects server from too much load and still allows serving limited number of minimal delay requests if required.

General Recommendations:

It is not recommended to send very large requests (> 3MB) or attempt to update very long lists of objects (> 5000) in one request as the server may fail to fulfill such requests due to internal restrictions on memory consumption and execution time.

2.5 Supported Methods

In this version seven following methods are supported

```
areaCodes.listRegions

customers.listCustomers
customers.addCustomer
customers.updateCustomer
customers.addPayment
customers.deleteCustomer

equipment.listGateways
equipment.addSIPEquipment
equipment.addH323Equipment
equipment.updateSIPEquipment
equipment.updateH323Equipment
equipment.deleteEquipment

routing.listRoutes
routing.addRoutes
routing.updateRoutes
routing.deleteRoutes

cdr.listSuccessCalls
cdr.listFailedCalls
cdr.listAggregate
cdr.listSummary
```

They are described in details below.

3 Methods

3.1.1 areaCodes.listRegions

Used to make different queries on Hostedswitch region tree. It accepts the following request parameters

```
regionCode      - STRING OPTIONAL
                 Code (prefix of the phone number in E.164 format) to match against.
                 Only uniform numeric codes are valid, no ranges, comma or otherwise separated lists
                 will have any effect.

regionName      - STRING OPTIONAL
                 Full or part of region name as it appears in Hostedswitch web interface.

idList          - ARRAY OPTIONAL
                 An array of Hostedswitch regionId's (internal unique Hostedswitch region ID codes). Each regionId
                 is numeric but it is recommended to pass it as a string (of 1-10 decimal digits) so as to
                 avoid possible integer overflow.

recursiveCodes  - OPTIONAL
                 This parameter does not require a value, thus it is recommended to pass it empty.
                 If present, "codes" member of each of regions returned will contain codes for the region itself
                 and for all its child regions merged altogether. Otherwise "codes" will only contain codes that
                 belong to that very region with subregions' codes excluded.
```

Response parameters consist of a single "regionList" which is an array

```
regionList := ARRAY OF region
```

```

region      := STRUCT {
    MEMBER regionId      - STRING ALWAYS PRESENT
                        Hostedswitch internal unique ID of the region.

    MEMBER name          - STRING ALWAYS PRESENT
                        Region name as it appears in Hostedswitch web interface.

    MEMBER codes         - STRING ALWAYS PRESENT
                        List of codes (see below).

    MEMBER parentId      - STRING ALWAYS PRESENT
                        regionId of the parent region to the current one in the region tree.
                        May be empty if the current region is one of the root regions.

    MEMBER hasChildren   - BOOL ALWAYS PRESENT
                        Tells whether current region has subregions or not.
}

codes      := COMMA-SEPARATED LIST OF interval

interval   := codeStart "-" codeEnd | code
            Single range of codes with start and end codes separated by dash. If the start and end
            codes coincide then only one will be present.

```

See Appendix B.1 for example of this method call.

3.2.1 customers.listCustomers

Used to obtain information about registered customers. Accepts the following request parameters

```

customerId  - STRING OPTIONAL (1-10 decimal digits)
            Hostedswitch internal unique numeric ID of a Customer.

name        - STRING OPTIONAL (10 characters max)
            Name of the Customer. When specified at most one record is returned that corresponds to
            the Customer with that exact name.

status      - INTEGER OPTIONAL
            Status of the Customer. Following values are accepted only:
            100 - Registered
            150 - Disabled Origination
            200 - Enabled
            250 - Testing
            400 - Disabled

```

Statuses 100 and 250 exist for historical reasons and are not supported within current Hostedswitch web-interface or software.

Hostedswitch Customers can be thought of as groups of gateways that may be billed and managed separately. In API calls they can be addressed by numeric internal "customerId" or by name as it appears in Hostedswitch web interface (referred in API calls to as "customer"). This name is 10 characters long at most and it is unique per Hostedswitch account.

Response parameters consist of a single "customerList" which is an array

```

customerList := ARRAY OF customer

customer     := STRUCT {

    MEMBER customerId  - STRING ALWAYS PRESENT
                    Internal ID of the Customer.

    MEMBER name        - STRING ALWAYS PRESENT
                    Name of the Customer.

    MEMBER description - STRING ALWAYS PRESENT
                    Description of the customer.

    MEMBER default     - INTEGER ALWAYS PRESENT
                    Flag that shows whether this Customer is a default one.
                    1 - default
                    0 - not default

    MEMBER status      - INTEGER ALWAYS PRESENT
                    Status of the Customer. See "status" parameter of the request for this method for details.

```

```

MEMBER balance      - DOUBLE ALWAYS PRESENT
    Balance of the Customer. Balances are not updated in real-time, see "changeDate" parameter
    for the balance actuality.

MEMBER creditLine   - DOUBLE ALWAYS PRESENT
    Credit line given to the Customer.

MEMBER changeDate   - TIMESTAMP ALWAYS PRESENT
    Timestamp of the last update to the Customer record.
}

```

See Appendix B.2 for an example of this method call.

3.2.2 customers.addCustomer

Used to add new Customer. Accepts the following request parameters

```

name                - STRING REQUIRED (10 characters max)
    Unique name for a Customer

description         - STRING REQUIRED (100 characters max)
    Unique description for a Customer

creditLine          - DOUBLE REQUIRED (9999999 max)
    Customer Credit Line (to limit Customer's originated traffic)

settlementPeriod   - INTEGER REQUIRED
    Period for invoices automatic generation
    Following values are accepted only:
    0 - no invoices automatic generation
    1 - per half a month (1st day of month and 15th day of month)
    2 - per month (1st day of month)
    3 - per 2 months
    4 - every week on Monday
    5 - every week on Tuesday
    6 - every week on Wednesday
    7 - every week on Thursday
    8 - every week on Friday
    9 - every week on Saturday
    10 - every week on Sunday
    11 - every day
    12 - every 2 days
    13 - every 3 days
    14 - every 4 days
    15 - every 5 days
    16 - every 6 days

webPassword        - STRING OPTIONAL (6 characters min, 20 characters max)
    Password to enable Customer login (Customer's site functionality)

emails             - ARRAY OF STRING OPTIONAL
    Customer's emails array (Customer invoices may be sent to these emails)

inclBankInfo       - BOOL OPTIONAL
    Whether to show account Bank Info in Customer's invoices

autosend           - BOOL OPTIONAL
    Whether to send automatically generated invoices to a Customer

detalization       - INTEGER REQUIRED
    Invoices detalization level
    Following values are accepted only:
    1 - show traffic detalization in an invoice
    2 - show only totals in an invoice

invoiceType        - INTEGER REQUIRED
    Invoices type
    Following values are accepted only:
    1 - show full invoice with start, end balances
    2 - show only billed differences for the invoice billing period

invoiceFooter      - STRING OPTIONAL (4000 characters max)
    Custom footer to show in every Customer's invoice

```

When OPTIONAL request parameter is not passed, corresponding field is set to the default value represented in the description in angle brackets after keyword DEFAULT. If keyword DEFAULT does not exists then empty value is passed.

Response contains single parameter

status - INTEGER ALWAYS PRESENT

Status of the requested operation. Currently only 0 can be returned that denotes success.

3.2.3 customers.updateCustomer

Used to make alterations to Customer. Accepts the following request parameters

customerId - STRING REQUIRED (10 characters max)
Internal ID of the Customer.

name - STRING OPTIONAL (10 characters max)
Unique name for a Customer

description - STRING OPTIONAL (100 characters max)
Unique description for a Customer

origLimit - DOUBLE OPTIONAL (9999999 max)
Customer Credit Line (to limit Customer's originated traffic)

custStatus - INTEGER OPTIONAL
Customer status
Following values are accepted only:
200 - enabled
150 - origination disabled
400 - disabled

settlementPeriod - INTEGER OPTIONAL
Period for invoices automatic generation
Following values are accepted only:
0 - no invoices automatic generation
1 - per half a month (1st day of month and 15th day of month)
2 - per month (1st day of month)
3 - per 2 months
4 - every week on Monday
5 - every week on Tuesday
6 - every week on Wednesday
7 - every week on Thursday
8 - every week on Friday
9 - every week on Saturday
10 - every week on Sunday
11 - every day
12 - every 2 days
13 - every 3 days
14 - every 4 days
15 - every 5 days
16 - every 6 days

webPassword - STRING OPTIONAL (6 characters min, 20 characters max)
Password to enable Customer login (Customer's site functionality)

emails - ARRAY of STRING OPTIONAL
Customer's emails array (Customer invoices may be sent to these emails)

inclBankInfo - BOOL OPTIONAL
Whether to show account Bank Info in Customer's invoices

autosend - BOOL OPTIONAL
Whether to send automatically generated invoices to a Customer

detailization - INTEGER REQUIRED
Invoices detailization level
Following values are accepted only:
1 - show traffic detailization in an invoice
2 - show only totals in an invoice

invoiceType - INTEGER REQUIRED
Invoices type
Following values are accepted only:
1 - show full invoice with start, end balances
2 - show only billed differences for the invoice billing period

invoiceFooter - STRING OPTIONAL (4000 characters max)
Custom footer to show in every Customer's invoice

When OPTIONAL request parameter is not passed, corresponding field is set to the default value represented in the description in angle brackets after keyword DEFAULT. If keyword DEFAULT does not exist then empty value is passed.

Response contains single parameter

status - INTEGER ALWAYS PRESENT
Status of the requested operation. Currently only 0 can be returned that denotes success.

3.2.4 customers.addPayment

Used to add Payment to a Customer. Accepts the following request parameters

customerId - STRING REQUIRED (10 characters max)
Internal ID of the Customer.

description - STRING OPTIONAL (100 characters max)
Payment description

amount - DOUBLE REQUIRED (9999999 max)
Payment amount

from - INTEGER REQUIRED
Payment direction
Following values are accepted only:
1 - from Customer to Account
-1 - from Account to Customer

Response contains single parameter

status - INTEGER ALWAYS PRESENT
Status of the requested operation. Currently only 0 can be returned that denotes success.

3.2.5 customers.deleteCustomer

Used to delete Customer that is of no use anymore.
Accepts the following request parameter

customerId - STRING REQUIRED (10 characters max)
Internal ID of the Customer.

Response contains single parameter

status - INTEGER ALWAYS PRESENT
Status of the requested operation. Currently only 0 can be returned that denotes success.

3.3.1 equipment.listGateways

Used to search for equipment registered in Hostedswitch. Accepts the following request parameters

ipAddress - STRING OPTIONAL
Full or partial gateway's IP address in dotted format.

username - STRING OPTIONAL (32 characters max)
Full or partial authorization username of a SIP gateway.

customerId - STRING OPTIONAL (1-10 decimal digits)
Hostedswitch internal unique numeric ID of a Customer, associated with a gateway.

customer - STRING OPTIONAL (10 characters max)
Name of the Customer, associated with a gateway.

tagList - ARRAY OPTIONAL
An array of User Tags, associated with gateways (see below). Each item is a STRING
64 characters long at most.

Particular registered gateway can potentially be addressed by internal ID as well but this ID is an object to change after certain update operations, hence we introduced a substitution for internal ID for API calls which we refer to as "User Tag" or just "tag". The tag is a member-defined string 64 characters long at most that can be used to uniquely identify an object (gateway, in particular). The tag can be specified upon creation of a gateway and can be changed at anytime via Hostedswitch web interface. If the tag is not specified it is assigned automatically to a value derived from internal ID

auto-tag := "###" internalId

Prefix "###" was selected so as to auto-assigned tags are unlikely to coincide with the tags supplied by member. Member-supplied tags cannot be longer than 64 characters and cannot contain non-printable or white-space characters (such as space, tab, linefeed, carriage return, vertical feed). The last requirement was introduced in order to avoid possible problems with human input of tags via web forms.

Response parameters consist of a single "gatewayList" which is an array

gatewayList := ARRAY OF gateway

gateway := STRUCT {

MEMBER tag - STRING ALWAYS PRESENT
Tag assigned to a gateway.

MEMBER gwId - STRING ALWAYS PRESENT
Hostedswitch internal gateway ID. It MAY change if the gateway is updated (thus, it will change if the gateway IP address is changed) hence it is not recommended to rely on.

MEMBER ipAddress - STRING ALWAYS PRESENT
IP address of a gateway in dotted format. For SIP gateways IP address may be unspecified, in this case "0.0.0.0" will be returned.

MEMBER ipRangeEnd - STRING ALWAYS PRESENT
Last IP address in the range. Non-empty only if the gateway is registered with the range of IP addresses.

MEMBER description - STRING ALWAYS PRESENT
Description of a gateway as it appears in Hostedswitch web interface.

MEMBER protocolId - INTEGER ALWAYS PRESENT
ID of the signaling protocol
1 - H.323 GW (Gateway)
2 - H.323 GK (Gatekeeper)
4 - SIP over UDP

MEMBER protocol - STRING ALWAYS PRESENT
Descriptive name of the signaling protocol.

MEMBER status - INTEGER ALWAYS PRESENT
Status of a gateway with respect to origination and termination:
1 - gateway is enabled for origination
2 - gateway is enabled for termination
3 - gateway is enabled both for origination and termination

MEMBER sswIp - STRING ALWAYS PRESENT
IP address of a softswitch dialpeer.

MEMBER customerId - STRING ALWAYS PRESENT
Internal ID of the Customer assigned to a gateway.

MEMBER customer - STRING ALWAYS PRESENT
Name of the Customer assigned to a gateway.

MEMBER prefixOrig - STRING ALWAYS PRESENT
Technical prefix for origination.

MEMBER prefixTerm - STRING ALWAYS PRESENT
Technical prefix for termination.

MEMBER portsOrig - INTEGER ALWAYS PRESENT
Number of ports (voice channels) for origination, 0 means unlimited.

MEMBER portsTerm - INTEGER ALWAYS PRESENT
Number of ports (voice channels) for termination.

MEMBER usernameOrig - STRING (PRESENT only for SIP gateways)
Authorization username of a SIP gateway (used for origination).

MEMBER passwordOrig - STRING (PRESENT only for SIP gateways)
Authorization password of a SIP gateway (used for origination).

MEMBER usernameTerm - STRING (PRESENT only for SIP gateways)
Authorization username of a SIP gateway (used for termination).

MEMBER passwordTerm - STRING (PRESENT only for SIP gateways)
Authorization password of a SIP gateway (used for termination).

MEMBER callModeOrig - INTEGER (PRESENT only for H.323 gateways)
ID of the H.323 call mode a gateway supports for origination:
1 - Slow Start
2 - Fast Start without H.245 Tunneling
3 - Slow and Fast Start without H.245 Tunneling
6 - Fast Start with H.245 Tunneling

7 - Slow and Fast Start with H.245 Tunneling

```
MEMBER callModeTerm - INTEGER (PRESENT only for H.323 gateways)
  ID of the H.323 call mode a gateway supports for termination
  1 - Slow Start
  2 - Fast Start without H.245 Tunneling
  3 - Slow and Fast Start without H.245 Tunneling
  6 - Fast Start with H.245 Tunneling
  7 - Slow and Fast Start with H.245 Tunneling
}
```

See Appendix B.3 for an example of this method call.

3.3.2 equipment.addSIPequipment

Used to add new SIP equipment. Accepts the following request parameters

```
description      - STRING REQUIRED (100 characters max)
  Unique description for an equipment

ipAddress        - STRING REQUIRED
  Full equipment's IP address in dotted format. (Start IP address of an IP range
  for an equipment authenticated by IP addresses range)

customerId       - STRING REQUIRED (1-10 decimal digits)
  Hostedswitch internal unique numeric ID of a Customer, associated with a gateway.

sipAuthType      - INTEGER REQUIRED (32 characters max)
  SIP equipment authentication type. Accepted values:
  1 - by username, password
  2 - by equipment IP address
  3 - by range of equipment IP addresses

endIpAddress     - STRING OPTIONAL
  End IP address of an IP range for an equipment authenticated by IP addresses range.

prefixOrig       - STRING OPTIONAL (20 characters max)
  Technical prefix to be truncated from a phone number when orig call is coming from this
  exact equipment.

prefixTerm       - STRING OPTIONAL (10 characters max)
  Technical prefix to be added to a phone number when a call is proxied to this
  exact equipment.

dnis             - STRING OPTIONAL (20 characters max)
  Technical prefix that used to separate equipment with same IP addresses by prefix.

usernameOrig     - STRING (32 characters max)
  Authorization username of a SIP gateway (used for origination). REQUIRED parameter when
  authorization by username is used.

passwordOrig     - STRING (32 characters max)
  Authorization password of a SIP gateway (used for origination). REQUIRED parameter when
  authorization by username is used.

usernameTerm     - STRING OPTIONAL (32 characters max)
  Authorization username of a SIP gateway (used for termination).

passwordTerm     - STRING OPTIONAL (32 characters max)
  Authorization password of a SIP gateway (used for termination).

tag              - STRING OPTIONAL (64 characters max) DEFAULT <"###" gwId>
  Tag assigned to a gateway.

portsOrig        - INTEGER OPTIONAL (9999 max)
  Number of ports (voice channels) for origination, 0 means unlimited.

portsTerm        - INTEGER OPTIONAL (9999 max) DEFAULT <120>
  Number of ports (voice channels) for termination.

csPort           - INTEGER OPTIONAL (65535 max) DEFAULT <5060>
  Termination equipment port.
```

When OPTIONAL request parameter is not passed, corresponding field is set to the default value represented in the description in angle brackets after keyword DEFAULT. If keyword DEFAULT does not exists then empty value is passed.

Response contains single parameter

status - INTEGER ALWAYS PRESENT

Status of the requested operation. Currently only 0 can be returned that denotes success.

3.3.3 equipment.addH323Equipment

Used to add new H323 equipment. Accepts the following request parameters

description - STRING REQUIRED (100 characters max)
Unique description for an equipment

ipAddress - STRING REQUIRED
Full equipment's IP address in dotted format. (Start IP address of an IP range for an equipment authenticated by IP addresses range)

customerId - STRING REQUIRED (1-10 decimal digits)
Hostedswitch internal unique numeric ID of a Customer, associated with a gateway.

endIpAddress - STRING OPTIONAL
End IP address of an IP range for an equipment authenticated by IP addresses range.

prefixOrig - STRING OPTIONAL (20 characters max)
Technical prefix to be truncated from a phone number when orig call is coming from this exact equipment.

prefixTerm - STRING OPTIONAL (10 characters max)
Technical prefix to be added to a phone number when a call is proxied to this exact equipment.

dnis - STRING OPTIONAL (20 characters max)
Technical prefix that used to separate equipment with same IP addresses by prefix. If dnis is used it MUST be equal either to prefixTruncate or to termPrefix.

callModeOrig - INTEGER OPTIONAL DEFAULT <7>
ID of the H.323 call mode a gateway supports for origination:
1 - Slow Start
2 - Fast Start without H.245 Tunneling
3 - Slow and Fast Start without H.245 Tunneling
6 - Fast Start with H.245 Tunneling
7 - Slow and Fast Start with H.245 Tunneling

callModeTerm - INTEGER OPTIONAL DEFAULT <7>
ID of the H.323 call mode a gateway supports for termination:
1 - Slow Start
2 - Fast Start without H.245 Tunneling
3 - Slow and Fast Start without H.245 Tunneling
6 - Fast Start with H.245 Tunneling
7 - Slow and Fast Start with H.245 Tunneling

tag - STRING OPTIONAL (64 characters max) DEFAULT <"###" gwId>
Tag assigned to a gateway.

portsOrig - INTEGER OPTIONAL (9999 max)
Number of ports (voice channels) for origination, 0 means unlimited.

portsTerm - INTEGER OPTIONAL (9999 max) DEFAULT <120>
Number of ports (voice channels) for termination.

csPort - INTEGER OPTIONAL (65535 max) DEFAULT <5060>
Termination equipment port.

When OPTIONAL request parameter is not passed, corresponding field is set to the default value represented in the description in angle brackets after keyword DEFAULT. If keyword DEFAULT does not exist then empty value is passed.

Response contains single parameter

status - INTEGER ALWAYS PRESENT

Status of the requested operation. Currently only 0 can be returned that denotes success.

3.3.4 equipment.updateSIPequipment

Used to make alterations to the SIP equipment only.

SIP equipment has protocolId = 4 - response parameter from "equipment.listGateways" method. Accepts the following request parameters

tag - STRING REQUIRED (64 characters max)
Tag to identify the gateway to be updated.

description - STRING OPTIONAL (100 characters max)

Unique description for an equipment

ipAddress - STRING OPTIONAL
Full equipment's IP address in dotted format. (Start IP address of an IP range for an equipment authenticated by IP addresses range)

customerId - STRING OPTIONAL (1-10 decimal digits)
Hostedswitch internal unique numeric ID of a Customer, associated with a gateway.

gwStatus - INTERGER OPTIONAL
Equipment status:
0 - equipment disabled
1 - origination allowed only
2 - termination allowed only
3 - origination and termination are allowed

sipAuthType - INTEGER OPTIONAL (32 characters max)
SIP equipment authentication type. Accepted values:
1 - by username, password
2 - by equipment IP address
3 - by range of equipment IP addresses

endIpAddress - STRING OPTIONAL
End IP address of an IP range for an equipment authenticated by IP addresses range.

prefixOrig - STRING OPTIONAL (20 characters max)
Technical prefix to be truncated from a phone number when orig call is coming from this exact equipment.

prefixTerm - STRING OPTIONAL (10 characters max)
Technical prefix to be added to a phone number when a call is proxied to this exact equipment.

dnis - STRING OPTIONAL (20 characters max)
Technical prefix that used to separate equipment with same IP addresses by prefix.
If dnis is used it MUST be equal either to prefixTruncate or to termPrefix.

usernameOrig - STRING OPTIONAL (32 characters max)
Authorization username of a SIP gateway (used for origination).

passwordOrig - STRING OPTIONAL (32 characters max)
Authorization password of a SIP gateway (used for origination).

usernameTerm - STRING OPTIONAL (32 characters max)
Authorization username of a SIP gateway (used for termination).

passwordTerm - STRING OPTIONAL (32 characters max)
Authorization password of a SIP gateway (used for termination).

portsOrig - INTEGER OPTIONAL (9999 max)
Number of ports (voice channels) for origination, 0 means unlimited.

portsTerm - INTEGER OPTIONAL (9999 max)
Number of ports (voice channels) for termination.

csPort - INTEGER OPTIONAL (65535 max)
Termination equipment port.

Most of the parameters are OPTIONAL; when present, the respective field is updated to the value provided within the member, otherwise the field is left unchanged.

Response contains single parameter

status - INTEGER ALWAYS PRESENT
Status of the requested operation. Currently only 0 can be returned that denotes success.

3.3.5 equipment.updateH323Equipment

Used to make alterations to the H323 equipment only.
H323 equipment has protocolId = 1 - response parameter from "equipment.listGateways" method.
Accepts the following request parameters

tag - STRING REQUIRED (64 characters max)
Tag to identify the gateway to be updated.

description - STRING OPTIONAL (100 characters max)
Unique description for an equipment

ipAddress - STRING OPTIONAL

Full equipment's IP address in dotted format. (Start IP address of an IP range for an equipment authenticated by IP addresses range)

customerId - STRING OPTIONAL (1-10 decimal digits)
Hostedswitch internal unique numeric ID of a Customer, associated with a gateway.

gwStatus - INTERGER OPTIONAL
Equipment status:
0 - equipment disabled
1 - origination allowed only
2 - termination allowed only
3 - origination and termination are allowed

endIpAddress - STRING OPTIONAL
End IP address of an IP range for an equipment authenticated by IP addresses range.

prefixOrig - STRING OPTIONAL (20 characters max)
Technical prefix to be truncated from a phone number when orig call is coming from this exact equipment.

prefixTerm - STRING OPTIONAL (10 characters max)
Technical prefix to be added to a phone number when a call is proxied to this exact equipment.

dnis - STRING OPTIONAL (20 characters max)
Technical prefix that used to separate equipment with same IP addresses by prefix.
If dnis is used it MUST be equal either to prefixTruncate or to termPrefix.

callModeOrig - INTEGER OPTIONAL
ID of the H.323 call mode a gateway supports for origination:
1 - Slow Start
2 - Fast Start without H.245 Tunneling
3 - Slow and Fast Start without H.245 Tunneling
6 - Fast Start with H.245 Tunneling
7 - Slow and Fast Start with H.245 Tunneling

callModeTerm - INTEGER OPTIONAL
ID of the H.323 call mode a gateway supports for termination:
1 - Slow Start
2 - Fast Start without H.245 Tunneling
3 - Slow and Fast Start without H.245 Tunneling
6 - Fast Start with H.245 Tunneling
7 - Slow and Fast Start with H.245 Tunneling

portsOrig - INTEGER OPTIONAL (9999 max)
Number of ports (voice channels) for origination, 0 means unlimited.

portsTerm - INTEGER OPTIONAL (9999 max)
Number of ports (voice channels) for termination.

csPort - INTEGER OPTIONAL (65535 max)
Termination equipment port.

Most of the parameters are OPTIONAL; when present, the respective field is updated to the value provided within the member, otherwise the field is left unchanged.

Response contains single parameter

status - INTEGER ALWAYS PRESENT
Status of the requested operation. Currently only 0 can be returned that denotes success.

3.3.6 equipment.deleteEquipment

Used to delete equipment that is of no use anymore.
Accepts the following request parameter

tag - STRING REQUIRED (64 characters max)
Tag to identify the gateway to be updated.

Response contains single parameter

status - INTEGER ALWAYS PRESENT
Status of the requested operation. Currently only 0 can be returned that denotes success.

3.4.1 routing.listRoutes

Used to obtain detailed information on established routes. Accepts the following request parameters

regionCode - STRING OPTIONAL
Code (prefix of the phone number in E.164 format) to match against.
Only uniform numeric codes are valid, no ranges, comma or otherwise separated lists will have any effect.

regionName - STRING OPTIONAL
Full or part of region name as it appears in Hostedswitch web interface.

routeIds - ARRAY OPTIONAL
An array of Hostedswitch routeId's (internal unique Hostedswitch route ID). Each routeId is numeric but it is recommended to pass it as a string (of 1-10 decimal digits) so as to avoid possible integer overflow.

tagList - ARRAY OPTIONAL
An array of tags, associated with routes (see below). Each item is a STRING 64 characters long at most.

customerIdOrig - STRING OPTIONAL (1-10 decimal digits)
Internal numeric ID of the originating Customer of the route to search.

customerOrig - STRING OPTIONAL (10 characters max)
Name of the originating Customer of the route to search.

customerIdTerm - STRING OPTIONAL (1-10 decimal digits)
Internal numeric ID of the terminating Customer of the route to search.

customerTerm - STRING OPTIONAL (10 characters max)
Name of the terminating Customer of the route to search.

gwTagOrig - STRING OPTIONAL (64 characters max)
Tag of the originating gateway of the route to search.

gwTagTerm - STRING OPTIONAL (64 characters max)
Tag of the terminating gateway of the route to search.

status - INTEGER OPTIONAL
Status of the route to search:
-1 - only disabled routes returned
1 - only active routes returned
0 or not present - all routes are returned
Other values are not acceptable.

rtpOn - INTEGER OPTIONAL
Status of the RTP proxy for the route to search:
-1 - only routes with RTP proxy off returned
1 - only routes with RTP proxy on returned
0 or not present - all routes returned
Other values are not acceptable.

endpointListType - INTEGER OPTIONAL
If present, this parameter determines how route endpoints (i.e. lists of originators and terminators) are returned:
1 - list of endpoints is returned as an ARRAY of customerId
2 - list of endpoints is returned as an ARRAY of customer (i.e. Customer name)
3 - list of endpoints is returned as an ARRAY of gwId (i.e. internal Hostedswitch Id of the gateway)
4 - list of endpoints is returned as an ARRAY of gwTag (i.e. tag of the gateway)
When not present, no endpoints are returned.

Response parameters consist of a single "routeList" which is an array

```
routeList := ARRAY OF route

route := STRUCT {

    MEMBER tag - STRING ALWAYS PRESENT
    Tag assigned to the route.

    MEMBER routeId - STRING ALWAYS PRESENT
    Hostedswitch internal route ID. It MAY change if the route is updated (thus, it will change if the price or billing settings are changed) hence it is not recommended to rely on.

    MEMBER regionId - STRING ALWAYS PRESENT
    ID of the region (see areaCodes.listRegion for details on regionId) the route is established to.

    MEMBER regionName - STRING ALWAYS PRESENT
    Name of the region the route is established to.
```

MEMBER tariffOrig - DOUBLE ALWAYS PRESENT
Tariff for origination (\$US per minute).

MEMBER tariffTerm - DOUBLE ALWAYS PRESENT
Tariff for termination (\$US per minute).

MEMBER billingIncOrig - INTEGER ALWAYS PRESENT
Billing Increment for origination.

MEMBER freeTimeOrig - INTEGER ALWAYS PRESENT
Grace Period for origination.

MEMBER minCallTimeOrig - INTEGER ALWAYS PRESENT
Minimum Duration for origination.

MEMBER billingIncTerm - INTEGER ALWAYS PRESENT
Billing Increment for termination.

MEMBER freeTimeTerm - INTEGER ALWAYS PRESENT
Grace Period for termination.

MEMBER minCallTimeTerm - INTEGER ALWAYS PRESENT
Minimum Duration for termination.

MEMBER priority - INTEGER ALWAYS PRESENT
Route Priority.

MEMBER weight - INTEGER ALWAYS PRESENT
Route Weight.

MEMBER numPorts - INTEGER ALWAYS PRESENT
Number of ports (channels) per route. This is the number expressed in units defined by "portType" member.

MEMBER portType - INTEGER ALWAYS PRESENT
Unit for "numPorts" parameter:
1 - E1
2 - T1
3 - Port

MEMBER status - INTEGER ALWAYS PRESENT
Route status:
0 - route is disabled
1 - route is active

MEMBER codecId - INTEGER ALWAYS PRESENT
ID of the codec set for the route:
10 - G.711
30 - G.723
70 - G.729 + G.723
110 - G.729

MEMBER codec - STRING ALWAYS PRESENT
Descriptive name of the codec set for the route.

MEMBER endpointListType - INTEGER (PRESENT only if "endpointListType" is passed in the request)
Determines what is returned in "originators" and "terminators" parameters. The value is the same as one passed in the request.

MEMBER originators - ARRAY (PRESENT only if "endpointListType" is passed in the request)
List of originating endpoints of the route. An array of elements to be interpreted according to "endpointListType" value (see description of this method's request).

MEMBER terminators - ARRAY (PRESENT only if "endpointListType" was passed in the request)
List of terminating endpoints of the route. An array of elements to be interpreted according to "endpointListType" value.

MEMBER dtRuleId - STRING (PRESENT only if the route has Daytime Rule associated)
Internal numeric ID of the Daytime Rule associated with the route.

MEMBER aniFiltering - INTEGER (PRESENT only if ANI Filtering feature is enabled for the account)
ANI Filtering state for the route flag:
0 - ANI Filtering is off
1 - ANI Filtering is on

MEMBER rtpOn - INTEGER (PRESENT only if RTP Proxy feature is enabled for the account)
RTP Proxy state for the route flag:
0 - RTP Proxy is off

```
    1 - RTP Proxy is on
}
```

Route tag has the same purpose and meaning as aforementioned gateway tag. For the meaning of Billing Increment, Grace Period, Minimum Duration, Priority, Weight, E1 and T1 see Hostedswitch Glossary

<https://hostedswitch.com/hs/?Glossary>

See Appendix B.4 for example of this method call.

3.4.2 routing.addRoutes

Used to established new routes. Accepts one parameter

`routes` := ARRAY of route, REQUIRED

An array of routes to be established. Each item is a "route" STRUCT (described below) which holds a set of parameters needed to construct a route.

`route` := STRUCT {

MEMBER tag - STRING OPTIONAL DEFAULT <"###" routeId>
Tag to be assigned to the new route. If not present, tag for the new route will be assigned automatically based on internal numeric route ID (see DEFAULT).

MEMBER regionId - STRING REQUIRED
ID of the region (see `areaCodes.listRegion` for details on regionId).

MEMBER tariffOrig - DOUBLE REQUIRED
Tariff for origination (\$US per minute).

MEMBER tariffTerm - DOUBLE REQUIRED
Tariff for termination (\$US per minute).

MEMBER billingIncOrig - INTEGER OPTIONAL DEFAULT <1>
Billing Increment for origination.

MEMBER freeTimeOrig - INTEGER OPTIONAL DEFAULT <0>
Grace Period for origination.

MEMBER minCallTimeOrig - INTEGER OPTIONAL DEFAULT <0>
Minimum Duration for origination.

MEMBER billingIncTerm - INTEGER OPTIONAL DEFAULT <1>
Billing Increment for termination.

MEMBER freeTimeTerm - INTEGER OPTIONAL DEFAULT <0>
Grace Period for termination.

MEMBER minCallTimeTerm - INTEGER OPTIONAL DEFAULT <0>
Minimum Duration for termination.

MEMBER priority - INTEGER OPTIONAL DEFAULT <1>
Route Priority. Accepted values:
1, 2, 3, 4, 5

MEMBER weight - INTEGER OPTIONAL DEFAULT <100>
Route Weight. Accepted values:
0, 3, 5, 10, 15, 20, 25, 30, 40, 50, 60, 70, 80, 90, 100
Value of 0 exists for historical reasons and is not recommended for use as it may lead to unexpected traffic distribution among routes.

MEMBER numPorts - INTEGER REQUIRED
Number of ports (channels) per route. Integer not greater than 99999.

MEMBER portType - INTEGER REQUIRED
Unit for "numPorts" parameter. Accepted values:
1 - for E1
2 - for T1
3 - for Port

MEMBER status - INTEGER OPTIONAL DEFAULT <1>
Route status. Accepted values:
0 - for disabled
1 - for active

MEMBER codecId - INTEGER OPTIONAL DEFAULT <70>

ID of the codec. Accepted values:

10 - for G.711
30 - for G.723
70 - for G.729 + G.723
110 - for G.729

MEMBER endpointListType - INTEGER REQUIRED

Determines how "originators" and "terminators" parameters should be interpreted.

Accepted values:

1 - endpoints list is an ARRAY of customerId
2 - endpoints list is an ARRAY of customer (i.e. Customer name)
4 - endpoints list is an ARRAY of gwTag (i.e. tag of the gateway)

Please note, that value 3 (ARRAY of internal gwId) is not accepted.

MEMBER originators - ARRAY REQUIRED

List of originating endpoints of the route. An array of elements to be interpreted according to "endpointListType" value. Each item must conform to the restriction of the respective type:

customerId - STRING of 1-10 decimal digits
customer - STRING of 1-10 characters
gwTag - STRING of 1-64 printable non-space characters

Each item must refer to registered gateway or customer. Gateways that are not allowed for origination by status WILL NOT be added to the route endpoints set.

MEMBER terminators - ARRAY REQUIRED

List of terminating endpoints of the route. An array of elements to be interpreted according to "endpointListType" value. See "originators" parameter above for restrictions on item values.

Gateways that are not allowed for termination by status WILL NOT be added to the route endpoints set.

MEMBER dtRuleId - STRING OPTIONAL DEFAULT NONE

Internal numeric ID of the Daytime Rule to be associated with the route. String of 1-10 decimal digits. Must refer to existing Daytime Rule.

MEMBER aniFiltering - INTEGER OPTIONAL DEFAULT <0>

ANI Filtering state for the route flag. Accepted values:

0, 1

MEMBER rtpOn - INTEGER OPTIONAL DEFAULT <0>

RTP Proxy state for the route flag. Accepted values:

0, 1

}

When OPTIONAL member of the "route" struct above is not passed, corresponding field is set to the default value represented in the description in angle brackets after keyword DEFAULT.

Response contains single parameter

status - INTEGER ALWAYS PRESENT

Status of the requested operation. Currently only 0 can be returned that denotes success.

See Appendix B.5 for example of this method call.

3.4.3 routing.updateRoutes

Used to make alterations to the established routes. Accepts the following request parameters

fastMatch - OPTIONAL

This parameter does not require a value and thus it is recommended to pass it empty.

If present, routes to update are matched only by "tag" from the "route" struct below ("fast match").

Otherwise "regionId" member of the "route" struct is required to be passed and both "tag" and "regionId" must match respective fields of the route to be updated. If they are not or if no "regionId" member found, fault is returned.

This approach allows to find and update routes in safer and less error-prone way. At the same time if the client software implementors are confident of their system, "fast match" mode can be used.

routes := ARRAY of route, REQUIRED

route := STRUCT {

MEMBER tag - STRING REQUIRED

Tag to identify the route to be updated.

MEMBER regionId - STRING

This member is required when "fastMatch" parameter is not present in the request and it must match regionId field of the route with User Tag "tag".

```

MEMBER tariffOrig      - DOUBLE OPTIONAL
MEMBER tariffTerm     - DOUBLE OPTIONAL
MEMBER billingIncOrig - INTEGER OPTIONAL
MEMBER freeTimeOrig   - INTEGER OPTIONAL
MEMBER minCallTimeOrig - INTEGER OPTIONAL
MEMBER billingIncTerm - INTEGER OPTIONAL
MEMBER freeTimeTerm   - INTEGER OPTIONAL
MEMBER minCallTimeTerm - INTEGER OPTIONAL
MEMBER priority       - INTEGER OPTIONAL
MEMBER weight         - INTEGER OPTIONAL
MEMBER numPorts       - INTEGER OPTIONAL
MEMBER portType       - INTEGER OPTIONAL
MEMBER status         - INTEGER OPTIONAL
MEMBER codecId        - INTEGER OPTIONAL
MEMBER endpointListType - INTEGER OPTIONAL
    Note however, that if any of "originators" or "terminators" members is present, this field
    is required.
MEMBER originators     - ARRAY OPTIONAL
MEMBER terminators     - ARRAY OPTIONAL
MEMBER aniFiltering    - INTEGER OPTIONAL
MEMBER rtpOn           - INTEGER OPTIONAL
}

```

Meaning of the members of the "route" struct is the same as for routing.AddRoutes method. Most of the members are OPTIONAL; when present, the respective field is updated to the value provided within the member, otherwise the field is left unchanged.

Response contains single parameter

```

status - INTEGER ALWAYS PRESENT
    Status of the requested operation. Currently only 0 can be returned that denotes success.

```

See Appendix B.6 for example of this method call.

3.4.4 routing.deleteRoutes

Used to delete established routes. Accepts the following request parameters

```

fastMatch - OPTIONAL
    This parameter does not require a value and thus it is recommended to pass it empty.
    If present, routes to delete are matched only by "tag" from the "route" struct below ("fast match").
    Otherwise "regionId" member of the "route" struct is required to be passed and both "tag" and
    "regionId" must match respective fields of the route to delete.

routes := ARRAY of route, REQUIRED

route := STRUCT {
    MEMBER tag - STRING REQUIRED
        Tag to identify the route to be deleted.
    MEMBER regionId - STRING
        This member is required when "fastMatch" parameter is not present in the request and it
        must match regionId field of the route with User Tag "tag".
}

```

Response contains single parameter

status - INTEGER ALWAYS PRESENT

Status of the requested operation. Currently only 0 can be returned that denotes success.

See Appendix B.7 for example of this method call.

3.5.1 cdr.listSuccessCalls

Used to obtain call detailed records list (successful calls). Accepts the following request parameters

startDate - TIMESTAMP REQUIRED
Find CDRs starting from this date

endDate - TIMESTAMP REQUIRED
Find CDRs till this date. Time window between startDate and endDate can not exceeds 7 days.

page - INTEGER OPTIONAL DEFAULT <1>
Page to be returned. Max records in one request is 1000. If amount of found records exceeds 1000 then the whole list is separated into pages. Each page can be requested separately.

phoneNum - STRING OPTIONAL (1-20 decimal digits)
Full or part of phone number

customerIdOrig - STRING OPTIONAL (1-10 decimal digits)
Internal numeric ID of the originating Customer

customerIdTerm - STRING OPTIONAL (1-10 decimal digits)
Internal numeric ID of the terminating Customer

gwTagOrig - STRING OPTIONAL (64 characters max)
Tag of the originating gateway

gwTagTerm - STRING OPTIONAL (64 characters max)
Tag of the terminating gateway

routeId - STRING OPTIONAL (64 characters max)
Internal unique Hostedswitch route ID

Response parameters:

```
cdrList := ARRAY OF cdr
  cdr := STRUCT {
    MEMBER callDate - TIMESTAMP ALWAYS PRESENT
      Timestamp of a call
    MEMBER phoneNum - STRING ALWAYS PRESENT
      E.164 phone number
    MEMBER routeId - STRING ALWAYS PRESENT
      Internal unique Hostedswitch route ID (see also routing.listRoutes)
    MEMBER region - STRING ALWAYS PRESENT
      Region name as it appears in Hostedswitch web interface
    MEMBER origCustName - STRING ALWAYS PRESENT
      Name of an originating Customer
    MEMBER origCustDesc - STRING ALWAYS PRESENT
      Description of an originating Customer
    MEMBER termCustName - STRING ALWAYS PRESENT
      Name of a terminating Customer
    MEMBER termCustDesc - STRING ALWAYS PRESENT
      Description of a terminating Customer
    MEMBER origGwIp - STRING ALWAYS PRESENT
      IP address of an originating gateway
    MEMBER origGwId - STRING ALWAYS PRESENT
      Hostedswitch internal ID of the originating gateway (see also equipment.listGateways)
    MEMBER origGwDesc - STRING ALWAYS PRESENT
      Description of a originating gateway
    MEMBER termGwIp - STRING ALWAYS PRESENT
      IP address of a terminating gateway
```

```

MEMBER termGwId          - STRING ALWAYS PRESENT
    Hostedswitch internal ID of the terminating gateway (see also equipment.listGateways)

MEMBER termGwDesc        - STRING ALWAYS PRESENT
    Description of a terminating gateway

MEMBER origTariff         - DOUBLE ALWAYS PRESENT
    Tariff for origination ($US per minute)

MEMBER termTariff         - DOUBLE ALWAYS PRESENT
    Tariff for termination ($US per minute)

MEMBER origSumBilled     - DOUBLE ALWAYS PRESENT
    Amount billed for origination ($US)

MEMBER termSumBilled     - DOUBLE ALWAYS PRESENT
    Amount billed for termination ($US)

MEMBER origTimeBilled    - DOUBLE ALWAYS PRESENT
    Time billed at origination side (seconds)

MEMBER termTimeBilled    - DOUBLE ALWAYS PRESENT
    Time billed at terminaion side (seconds)

MEMBER duration          - DOUBLE ALWAYS PRESENT
    Pure call duration with no billing settings applied (seconds)

MEMBER ani                - STRING ALWAYS PRESENT
    Caller number
}

count                    - INTEGER ALWAYS PRESENT
    Total records count

page                     - INTEGER ALWAYS PRESENT
    Current list page

pagesTotal               - INTEGER ALWAYS PRESENT
    Total pages in a list

```

3.5.2 cdr.listFailedCalls

Used to obtain call detailed records list (failed calls). Accepts the following request parameters

```

startDate                - TIMESTAMP REQUIRED
    Find CDRs starting from this date

endDate                  - TIMESTAMP REQUIRED
    Find CDRs till this date. Time window between startDate and endDate can not exceeds 7 days.

page                     - INTEGER OPTIONAL DEFAULT <1>
    Page to be returned. Max records in one request is 1000. If amount of found records exceeds 1000
    then the whole list is separated into pages. Each page can be requested separately.

phoneNumber              - STRING OPTIONAL (1-20 decimal digits)
    Full or part of phone number

customerIdOrig           - STRING OPTIONAL (1-10 decimal digits)
    Internal numeric ID of the originating Customer

customerIdTerm           - STRING OPTIONAL (1-10 decimal digits)
    Internal numeric ID of the terminating Customer

gwTagOrig                - STRING OPTIONAL (64 characters max)
    Tag of the originating gateway

gwTagTerm                - STRING OPTIONAL (64 characters max)
    Tag of the terminating gateway

routeId                  - STRING OPTIONAL (64 characters max)
    Internal unique Hostedswitch route ID

```

Response parameters:

```

cdrList := ARRAY OF cdr
    cdr := STRUCT {

```

```

MEMBER callDate      - TIMESTAMP ALWAYS PRESENT
    Timestamp of a call

MEMBER phoneNum     - STRING ALWAYS PRESENT
    E.164 phone number

MEMBER routeId      - STRING ALWAYS PRESENT
    Internal unique Hostedswitch route ID (see also routing.listRoutes)

MEMBER region       - STRING ALWAYS PRESENT
    Region name as it appears in Hostedswitch web interface

MEMBER origCustName - STRING ALWAYS PRESENT
    Name of an originating Customer

MEMBER origCustDesc - STRING ALWAYS PRESENT
    Description of an originating Customer

MEMBER termCustName - STRING ALWAYS PRESENT
    Name of a terminating Customer

MEMBER termCustDesc - STRING ALWAYS PRESENT
    Description of a terminating Customer

MEMBER origGwIp     - STRING ALWAYS PRESENT
    IP address of an originating gateway

MEMBER origGwId     - STRING ALWAYS PRESENT
    Hostedswitch internal ID of the originating gateway (see also equipment.listGateways)

MEMBER origGwDesc   - STRING ALWAYS PRESENT
    Description of a originating gateway

MEMBER termGwIp     - STRING ALWAYS PRESENT
    IP address of a terminating gateway

MEMBER termGwId     - STRING ALWAYS PRESENT
    Hostedswitch internal ID of the terminating gateway (see also equipment.listGateways)

MEMBER termGwDesc   - STRING ALWAYS PRESENT
    Description of a terminating gateway

MEMBER rvFinal      - STRING ALWAYS PRESENT
    Final Softswitch return code in call hunting process

MEMBER rvCurrent    - STRING ALWAYS PRESENT
    Softswitch return code for exact call

MEMBER isdnCause    - STRING ALWAYS PRESENT
    ISDN cause for H323 calls or SIP return code for SIP call

MEMBER duration     - DOUBLE ALWAYS PRESENT
    Call duration (seconds)

MEMBER ani          - STRING ALWAYS PRESENT
    Caller number
}

count              - INTEGER ALWAYS PRESENT
    Total records count

page               - INTEGER ALWAYS PRESENT
    Current list page

pagesTotal        - INTEGER ALWAYS PRESENT
    Total pages in a list

```

3.5.3 cdr.listAggregate

Used to obtain Aggregations list. Accepts the following request parameters

```

startDate          - TIMESTAMP REQUIRED
    Find CDRs starting from this date

endDate           - TIMESTAMP REQUIRED
    Find CDRs till this date. Time window between startDate and endDate can not exceeds 7 days.

page              - INTEGER OPTIONAL DEFAULT <1>

```

Page to be returned. Max records in one request is 1000. If amount of found records exceeds 1000 then the whole list is separated into pages. Each page can be requested separately.

phoneNumber - STRING OPTIONAL (1-20 decimal digits)
Full or part of phone number

customerIdOrig - STRING OPTIONAL (1-10 decimal digits)
Internal numeric ID of the originating Customer

customerIdTerm - STRING OPTIONAL (1-10 decimal digits)
Internal numeric ID of the terminating Customer

gwTagOrig - STRING OPTIONAL (64 characters max)
Tag of the originating gateway

gwTagTerm - STRING OPTIONAL (64 characters max)
Tag of the terminating gateway

routeId - STRING OPTIONAL (64 characters max)
Internal unique Hostedswitch route ID

aggregateFields - ARRAY of STRING OPTIONAL DEFAULT <REGION>
Fields used for aggregations. Accepted values:
CUST_ORIG - aggregate by originating customer
GW_ORIG - aggregate by originating gateway
CUST_TERM - aggregate by terminating customer
GW_TERM - aggregate by terminating gateway
ROUTE_ID - aggregate by routes
REGION - aggregate by regions

Response parameters:

aggrList := ARRAY OF statRecord

statRecord := STRUCT {

- MEMBER successCalls - INTEGER ALWAYS PRESENT
Count of successful calls
- MEMBER failedCalls - INTEGER ALWAYS PRESENT
Count of failed calls
- MEMBER attempts - INTEGER ALWAYS PRESENT
Attempts count
- MEMBER routeId - STRING ALWAYS PRESENT
Internal unique Hostedswitch route ID (see also routing.listRoutes)
- MEMBER pdd - DOUBLE ALWAYS PRESENT
Post dial delay
- MEMBER region - STRING ALWAYS PRESENT
Region name as it appears in Hostedswitch web interface
- MEMBER origCustName - STRING ALWAYS PRESENT
Name of an originating Customer
- MEMBER origCustDesc - STRING ALWAYS PRESENT
Description of an originating Customer
- MEMBER termCustName - STRING ALWAYS PRESENT
Name of a terminating Customer
- MEMBER termCustDesc - STRING ALWAYS PRESENT
Description of a terminating Customer
- MEMBER origGwIp - STRING ALWAYS PRESENT
IP address of an originating gateway
- MEMBER origGwId - STRING ALWAYS PRESENT
Hostedswitch internal ID of the originating gateway (see also equipment.listGateways)
- MEMBER origGwDesc - STRING ALWAYS PRESENT
Description of a originating gateway
- MEMBER termGwIp - STRING ALWAYS PRESENT
IP address of a terminating gateway
- MEMBER termGwId - STRING ALWAYS PRESENT

```

        Hostedswitch internal ID of the terminating gateway (see also equipment.listGateways)

MEMBER termGwDesc      - STRING ALWAYS PRESENT
    Description of a terminating gateway

MEMBER origTariff      - DOUBLE ALWAYS PRESENT
    Tariff for origination ($US per minute)

MEMBER termTariff      - DOUBLE ALWAYS PRESENT
    Tariff for termination ($US per minute)

MEMBER origSumBilled   - DOUBLE ALWAYS PRESENT
    Amount billed for origination ($US)

MEMBER termSumBilled   - DOUBLE ALWAYS PRESENT
    Amount billed for termination ($US)

MEMBER origTimeBilled  - DOUBLE ALWAYS PRESENT
    Time billed at origination side (seconds)

MEMBER termTimeBilled  - DOUBLE ALWAYS PRESENT
    Time billed at terminaion side (seconds)

MEMBER duration        - DOUBLE ALWAYS PRESENT
    Calls duration (seconds)

MEMBER asr             - DOUBLE ALWAYS PRESENT
    Answer-Seizure Ratio (%)

MEMBER acd             - DOUBLE ALWAYS PRESENT
    Average call duration (seconds)
}

count                  - INTEGER (PRESENT when page = 1)
    Total records count

page                   - INTEGER ALWAYS PRESENT
    Current list page

pagesTotal             - INTEGER (PRESENT when page = 1)
    Total pages in a list

totals                  - STRUCT (PRESENT when page = 1)
    Totals for the whole list
    STRUCT members are:
    {
        MEMBER duration        - DOUBLE ALWAYS PRESENT
            Call duration (seconds)

        MEMBER origSumBilled    - DOUBLE ALWAYS PRESENT
            Amount billed for origination ($US)

        MEMBER termSumBilled    - DOUBLE ALWAYS PRESENT
            Amount billed for termination ($US)

        MEMBER origTimeBilled   - DOUBLE ALWAYS PRESENT
            Time billed at origination side (seconds)

        MEMBER termTimeBilled   - DOUBLE ALWAYS PRESENT
            Time billed at terminaion side (seconds)

        MEMBER successCalls     - INTEGER ALWAYS PRESENT
            Count of successful calls

        MEMBER failedCalls      - INTEGER ALWAYS PRESENT
            Count of failed calls

        MEMBER attempts         - INTEGER ALWAYS PRESENT
            Attempts count

        MEMBER asr              - DOUBLE ALWAYS PRESENT
            Answer-Seizure Ratio (%)

        MEMBER acd              - DOUBLE ALWAYS PRESENT
            Average call duration (seconds)
    }
}

```

3.5.4 cdr.listSummary

Used to obtain Summary list. Accepts the following request parameters

startDate - TIMESTAMP REQUIRED
Find CDRs starting from this date

endDate - TIMESTAMP REQUIRED
Find CDRs till this date. Time window between startDate and endDate can not exceeds 7 days.

page - INTEGER OPTIONAL DEFAULT <1>
Page to be returned. Max records in one request is 1000. If amount of found records exceeds 1000 then the whole list is separated into pages. Each page can be requested separately.

phoneNum - STRING OPTIONAL (1-20 decimal digits)
Full or part of phone number

customerIdOrig - STRING OPTIONAL (1-10 decimal digits)
Internal numeric ID of the originating Customer

customerIdTerm - STRING OPTIONAL (1-10 decimal digits)
Internal numeric ID of the terminating Customer

gwTagOrig - STRING OPTIONAL (64 characters max)
Tag of the originating gateway

gwTagTerm - STRING OPTIONAL (64 characters max)
Tag of the terminating gateway

routeId - STRING OPTIONAL (64 characters max)
Internal unique Hostedswitch route ID

Response parameters:

```
summList := ARRAY OF statRecord

statRecord := STRUCT {

    MEMBER successCallsOrig - INTEGER ALWAYS PRESENT
        Count of originated successful calls

    MEMBER successCallsTerm - INTEGER ALWAYS PRESENT
        Count of terminated successful calls

    MEMBER custName        - STRING ALWAYS PRESENT
        Name of a Customer

    MEMBER custDesc        - STRING ALWAYS PRESENT
        Description of a Customer

    MEMBER origSumBilled   - DOUBLE ALWAYS PRESENT
        Amount billed for origination ($US)

    MEMBER termSumBilled   - DOUBLE ALWAYS PRESENT
        Amount billed for termination ($US)

    MEMBER origTimeBilled  - DOUBLE ALWAYS PRESENT
        Time billed at origination side (seconds)

    MEMBER termTimeBilled  - DOUBLE ALWAYS PRESENT
        Time billed at terminaion side (seconds)

    MEMBER origDuration    - DOUBLE ALWAYS PRESENT
        Originated calls duration (seconds)

    MEMBER termDuration    - DOUBLE ALWAYS PRESENT
        Terminated calls duration (seconds)

    MEMBER commission      - DOUBLE ALWAYS PRESENT
        Customer commission ($US)
}

count                - INTEGER (PRESENT when page = 1)
Total records count

page                 - INTEGER ALWAYS PRESENT
Current list page

pagesTotal           - INTEGER (PRESENT when page = 1)
Total pages in a list
```

```

totals          - STRUCT (PRESENT when page = 1)
Totals for the whole list
STRUCT members are:
{
  MEMBER successCallsOrig - INTEGER ALWAYS PRESENT
    Count of originated successful calls

  MEMBER successCallsTerm - INTEGER ALWAYS PRESENT
    Count of terminated successful calls

  MEMBER origSumBilled    - DOUBLE ALWAYS PRESENT
    Amount billed for origination ($US)

  MEMBER termSumBilled    - DOUBLE ALWAYS PRESENT
    Amount billed for termination ($US)

  MEMBER origTimeBilled   - DOUBLE ALWAYS PRESENT
    Time billed at origination side (seconds)

  MEMBER termTimeBilled   - DOUBLE ALWAYS PRESENT
    Time billed at terminaion side (seconds)

  MEMBER origDuration     - DOUBLE ALWAYS PRESENT
    Originated calls duration (seconds)

  MEMBER termDuration     - DOUBLE ALWAYS PRESENT
    Terminated calls duration (seconds)

  MEMBER commission      - DOUBLE ALWAYS PRESENT
    Customers commission ($US)
}

```

Appendix A - Release Notes

A.1 Release 1.01

A.1.1 Restrictions changes

New restriction added:

The API server will only serve one request in 5 seconds for one authenticated account. Subsequent request within 5 seconds will result in a "Only 1 API request per 5 seconds is allowed" fault.

A.1.2 Methods changes

New methods added:

- customers.listCustomers
- customers.addCustomer
- customers.updateCustomer
- customers.addPayment
- customers.deleteCustomer

- equipment.listGateways
- equipment.addSIPEquipment
- equipment.addH323Equipment
- equipment.updateSIPEquipment
- equipment.updateH323Equipment
- equipment.deleteEquipment

- cdr.listSuccessCalls
- cdr.listFailedCalls
- cdr.listAggregate
- cdr.listSummary

A.1.3 Fixes

- Method "routing.listRoutes": regionCode, regionName parameters are working properly when filtering routes list

A.2 Release 1.02

A.2.1 Restrictions changes

Restriction on request rate (previously "one request in 5 seconds") is superseded with more flexible and convenient per-method rate tracking. See chapter 2.4

A.2.2 Method changes

CDR methods disabled until made consistent with equipment and route tags

A.3 Release 1.03

A.3.1 Method changes

CDR methods enabled.

To reflect use of internal Hostedswitch route and gateway identifiers

in these methods, following changes made:

- gwId member added to the output route struct of equipment.listGateways
- routeId member added to the output of routing.listRoutes
- endpointListType input parameter of routing.listRoutes now accepts value of 3 (ARRAY of gwId)
- origGwId and termGwId members added to the output route struct of cdr.listSuccessCalls, cdr.listFailedCalls and cdr.listAggregate

Appendix B - Requests Examples

B.1 areaCodes.listRegions example

Request

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<methodCall>
<methodName>areaCodes.listRegions</methodName>
<params>
<param>
<value>
<struct>
<member><name>.authMethod</name><value><int>1</int></value></member>
<member><name>.login</name><value><string>HS12345</string></value></member>
<member><name>.password</name><value><string>secret</string></value></member>
<member><name>idList</name>
<value>
<array>
<data>
<value><string>123456789</string></value>
<value><string>987654321</string></value>
</data>
</array>
</value>
</member>
<member><name>recursiveCodes</name><value/></member>
</struct>
</value>
</param>
</params>
</methodCall>
```

Response

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<methodResponse>
<params>
<param>
<value>
<struct>
<member>
<name>regionList</name>
<value>
<array>
<data>
<value>
<struct>
<member><name>regionId</name><value><string>123456789</string></value></member>
<member><name>name</name><value><string>Brazil</string></value></member>
<member><name>codes</name><value><string>5500-5555, 55600000-55656000, 556561-558888, 55889-55986, 559881, 559888-559999</string></value></member>
<member><name>parentId</name><value><string>123456788</string></value></member>
<member><name>hasChildren</name><value><boolean>1</boolean></value></member>
</struct>
</value>
</array>
</value>
<struct>
<member><name>regionId</name><value><string>987654321</string></value></member>
<member><name>name</name><value><string>Brazil, Brasilia</string></value></member>
<member><name>codes</name><value><string>55610-55616</string></value></member>
<member><name>parentId</name><value><string>987654310</string></value></member>
<member><name>hasChildren</name><value><boolean>0</boolean></value></member>
</struct>
</value>
</param>
</params>
```

```

    </data>
  </array>
</value>
</member>
</struct>
</value>
</param>
</params>
</methodResponse>

```

B.2 customers.listCustomers example

Request

```

<?xml version="1.0" encoding="iso-8859-1"?>
<methodCall>
<methodName>customers.listCustomers</methodName>
<params>
<param>
<value>
<struct>
<member><name>.authMethod</name><value><int>1</int></value></member>
<member><name>.login</name><value><string>HS12345</string></value></member>
<member><name>.password</name><value><string>secret</string></value></member>
<member><name>name</name><value><string>DCustomer</string></value></member>
</struct>
</value>
</param>
</params>
</methodCall>

```

Response

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<methodResponse>
<params>
<param>
<value>
<struct>
<member>
<name>customerList</name>
<value>
<array>
<data>
<value>
<struct>
<member><name>customerId</name><value><string>1649119567</string></value></member>
<member><name>name</name><value><string>DCustomer</string></value></member>
<member><name>description</name><value><string>Default</string></value></member>
<member><name>default</name><value><int>1</int></value></member>
<member><name>status</name><value><int>200</int></value></member>
<member><name>balance</name><value><double>1438.0132234</double></value></member>
<member><name>creditLine</name><value><double>0</double></value></member>
<member><name>changeDate</name><value><dateTime.iso8601>20070115T20:17:07</dateTime.iso8601></
value></member>
</struct>
</value>
</data>
</array>
</value>
</member>
</struct>
</value>
</param>
</params>
</methodResponse>

```

B.3 equipment.listGateways example

Request

```

<?xml version="1.0" encoding="iso-8859-1"?>
<methodCall>
<methodName>equipment.listGateways</methodName>
<params>
<param>
<value>

```

```

<struct>
  <member><name>.authMethod</name><value><int>1</int></value></member>
  <member><name>.login</name><value><string>HS12345</string></value></member>
  <member><name>.password</name><value><string>secret</string></value></member>
  <member><name>ipAddress</name><value><string>21.</string></value></member>
  <member>
    <name>tagList</name>
    <value>
      <array>
        <data>
          <value><string>qwerty</string></value>
        </data>
      </array>
    </value>
  </member>
</struct>
</value>
</param>
</params>
</methodCall>

```

Response

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<methodResponse>
<params>
<param>
<value>
<struct>
<member>
<name>gatewayList</name>
<value>
<array>
<data>
<value>
<struct>
<member><name>tag</name><value><string>qwerty</string></value></member>
<member><name>gwId</name><value><string>125345346</string></value></member>
<member><name>ipAddress</name><value><string>12.21.12.21</string></value></member>
<member><name>ipRangeEnd</name><value><string/></value></member>
<member><name>description</name><value><string>My GW</string></value></member>
<member><name>protocolId</name><value><int>1</int></value></member>
<member><name>protocol</name><value><string>H.323</string></value></member>
<member><name>status</name><value><int>3</int></value></member>
<member><name>sswIp</name><value><string>10.0.0.1</string></value></member>
<member><name>customerId</name><value><string>123456789</string></value></member>
<member><name>customer</name><value><string>DCustomer</string></value></member>
<member><name>prefixOrig</name><value><string/></value></member>
<member><name>prefixTerm</name><value><string>#777</string></value></member>
<member><name>portsOrig</name><value><int>0</int></value></member>
<member><name>portsTerm</name><value><int>120</int></value></member>
<member><name>callModeOrig</name><value><int>7</int></value></member>
<member><name>callModeTerm</name><value><int>7</int></value></member>
</struct>
</value>
</data>
</array>
</value>
</member>
</struct>
</value>
</param>
</params>
</methodResponse>

```

B.4 routing.listRoutes example

Request

```

<?xml version="1.0" encoding="iso-8859-1"?>
<methodCall>
<methodName>routing.listRoutes</methodName>
<params>
<param>
<value>
<struct>
<member><name>.authMethod</name><value><int>1</int></value></member>

```

```

<member><name>.login</name><value><string>HS12345</string></value></member>
<member><name>.password</name><value><string>secret</string></value></member>
<member>
  <name>tagList</name>
  <value>
    <array>
      <data>
        <value><string>###165954855</string></value>
      </data>
    </array>
  </value>
</member>
</struct>
</value>
</param>
</params>
</methodCall>

```

Response

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<methodResponse>
<params>
<param>
  <value>
    <struct>
      <member>
        <name>routeList</name>
        <value>
          <array>
            <data>
              <value>
                <struct>
                  <member><name>tag</name><value><string>###165954855</string></value></member>
                  <member><name>routeId</name><value><string>165954855</string></value></member>
                  <member><name>regionId</name><value><string>123456789</string></value></member>
                  <member><name>regionName</name><value><string>Afganistan</string></value></member>
                  <member><name>tariffOrig</name><value><double>0.2</double></value></member>
                  <member><name>tariffTerm</name><value><double>0.18</double></value></member>
                  <member><name>billingIncOrig</name><value><int>1</int></value></member>
                  <member><name>freeTimeOrig</name><value><int>0</int></value></member>
                  <member><name>minCallTimeOrig</name><value><int>0</int></value></member>
                  <member><name>billingIncTerm</name><value><int>1</int></value></member>
                  <member><name>freeTimeTerm</name><value><int>0</int></value></member>
                  <member><name>minCallTimeTerm</name><value><int>40</int></value></member>
                  <member><name>priority</name><value><int>1</int></value></member>
                  <member><name>weight</name><value><int>20</int></value></member>
                  <member><name>numPorts</name><value><int>21</int></value></member>
                  <member><name>portType</name><value><int>1</int></value></member>
                  <member><name>status</name><value><int>1</int></value></member>
                  <member><name>codecId</name><value><int>70</int></value></member>
                  <member><name>codec</name><value><string>G.729 + G.723</string></value></member>
                  <member><name>dtRuleId</name><value><string/></value></member>
                </struct>
              </value>
            </data>
          </array>
        </value>
      </struct>
    </value>
  </param>
</params>
</methodResponse>

```

B.5 routing.addRoutes example

Request

```

<?xml version="1.0" encoding="iso-8859-1"?>
<methodCall>
<methodName>routing.addRoutes</methodName>
<params>
<param>
  <value>
    <struct>
      <member><name>.authMethod</name><value><int>1</int></value></member>

```

```

<member><name>.login</name><value><string>HS12345</string></value></member>
<member><name>.password</name><value><string>secret</string></value></member>
<member>
  <name>routes</name>
  <value>
    <array>
      <data>
        <value>
          <struct>
            <member><name>regionId</name><value><string>123456789</string></value></member>
            <member><name>numPorts</name><value><int>20</int></value></member>
            <member><name>portType</name><value><int>1</int></value></member>
            <member><name>tariffOrig</name><value><double>0.2</double></value></member>
            <member><name>tariffTerm</name><value><double>0.18</double></value></member>
            <member><name>endpointListType</name><value><int>2</int></value></member>
            <member>
              <name>originators</name>
              <value>
                <array>
                  <data>
                    <value><string>cust#1</string></value>
                    <value><string>johndoe</string></value>
                  </data>
                </array>
              </value>
            </member>
            <member>
              <name>terminators</name>
              <value>
                <array>
                  <data>
                    <value><string>cust#2</string></value>
                  </data>
                </array>
              </value>
            </member>
            <member><name>tag</name><value><string>rt314159</string></value></member>
            <member><name>codecId</name><value><int>70</int></value></member>
            <member><name>billingIncOrig</name><value><int>6</int></value></member>
            <member><name>freeTimeOrig</name><value><int>30</int></value></member>
            <member><name>minCallTimeOrig</name><value><int>0</int></value></member>
          </struct>
        </value>
      </data>
    </array>
  </value>
</member>
</struct>
</value>
</param>
</params>
</methodCall>

```

Response

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<methodResponse>
<params>
  <param>
    <value>
      <struct>
        <member><name>status</name><value><int>0</int></value></member>
      </struct>
    </value>
  </param>
</params>
</methodResponse>

```

B.6 routing.updateRoutes example

Request

```

<?xml version="1.0" encoding="iso-8859-1"?>
<methodCall>
<methodName>routing.updateRoutes</methodName>
<params>
  <param>

```

```

<value>
  <struct>
    <member><name>.authMethod</name><value><int>1</int></value></member>
    <member><name>.login</name><value><string>HS12345</string></value></member>
    <member><name>.password</name><value><string>secret</string></value></member>
    <member>
      <name>routes</name>
      <value>
        <array>
          <data>
            <value>
              <struct>
                <member><name>tag</name><value><string>###1634534534</string></value></member>
                <member><name>regionId</name><value><string>123456789</string></value></member>
                <member><name>tariffOrig</name><value><double>0.22</double></value></member>
                <member><name>tariffTerm</name><value><double>0.19</double></value></member>
                <member><name>endpointListType</name><value><int>4</int></value></member>
                <member>
                  <name>originators</name>
                  <value>
                    <array>
                      <data>
                        <value><string>gw1422</string></value>
                        <value><string>gw2331</string></value>
                        <value><string>###134432473</string></value>
                      </data>
                    </array>
                  </value>
                </member>
                <member>
                  <name>terminators</name>
                  <value>
                    <array>
                      <data>
                        <value><string>###134432473</string></value>
                      </data>
                    </array>
                  </value>
                </member>
              </struct>
            </value>
          </data>
          <value>
            <struct>
              <member><name>tag</name><value><string>###1634534577</string></value></member>
              <member><name>regionId</name><value><string>123456789</string></value></member>
              <member><name>tariffOrig</name><value><double>0.21</double></value></member>
            </struct>
          </value>
        </data>
      </array>
    </value>
  </member>
</struct>
</value>
</params>
</methodCall>

```

Response

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<methodResponse>
<params>
  <param>
    <value>
      <struct>
        <member><name>status</name><value><int>0</int></value></member>
      </struct>
    </value>
  </param>
</params>
</methodResponse>

```

B.7 routing.deleteRoutes example

Request

```

<?xml version="1.0" encoding="iso-8859-1"?>

```

```

<methodCall>
<methodName>routing.deleteRoutes</methodName>
<params>
<param>
<value>
<struct>
<member><name>.authMethod</name><value><int>1</int></value></member>
<member><name>.login</name><value><string>HS12345</string></value></member>
<member><name>.password</name><value><string>secret</string></value></member>
<member><name>fastMatch</name><value><string/></value></member>
<member>
<name>routes</name>
<value>
<array>
<data>
<value>
<struct>
<member><name>tag</name><value><string>rt1023</string></value></member>
</struct>
</value>
<value>
<struct>
<member><name>tag</name><value><string>rt1024</string></value></member>
</struct>
</value>
<value>
<struct>
<member><name>tag</name><value><string>rt1025</string></value></member>
</struct>
</value>
</data>
</array>
</value>
</member>
</struct>
</value>
</param>
</params>
</methodCall>

```

Response

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<methodResponse>
<params>
<param>
<value>
<struct>
<member><name>status</name><value><int>0</int></value></member>
</struct>
</value>
</param>
</params>
</methodResponse>

```