

Call Analytics JSON-RPC API Guide

Version 20.10

CONTENTS

CHAPTER 1 About the Call Analytics JSON-RPC API

Overview	8
Requirements	9
JSON-RPC Support	9
Call Analytics API Server Limitations	9
API User Account and Permissions	9
Limitations of the Call Analytics API	10

CHAPTER 2 Call Analytics Key Concepts and Terminology

Terminology Quick Reference	12
Key Concepts	14
Accounts	14
Ad Campaigns	14
Ad Groups	15
Call Log	15
Call Tracking Numbers	16
Rewrites	16
Users	17

CHAPTER 3 Getting Started

JSON	18
JSON-RPC:	18
JSON syntax checkers:	18
Endpoint	18
Basic Access Authentication	19
HTTP Header Fields	19
Request Syntax	19
Request Parameters	20
Method Names	20
IDs	20
Send a POST Request via curl	21
Using Postman to Send a Request	21
Task List	21
Procedures	21

Install Chrome and Postman	21
Set the end point	22
Set the authentication method and credentials	22
Set the HTTP headers	22
Post a request to acct.list	22
Using Python to Send a Request	23
Downloading Python and pip	24
The Complete Example Code	25
A Close Look at the Example Code	26
Using PHP to Send a Request	28
Other HTTP Clients	28
The Complete Example Code	28
A Close Look at the Example Code	30

CHAPTER 4 Example Code for JSON-RPC API

Python Example Code	34
PHP Example Code	34

CHAPTER 5 JSON-RPC Reference

HTTP Status Codes	37
Accounts	37
acct.category.get	38
acct.category.set	40
acct.custom.define	41
acct.custom.get	45
acct.custom.set	49
acct.customid.get	51
acct.customid.set	52
acct.keyword_spotting.word_list.get	54
acct.keyword_spotting.word_list.set	57
acct.list	59
acct.membership.list	61
acct.membership.set	63
acct.membership.unset	65
acct.name.get	66
acct.name.set	67
acct.numlookup.set	69
acct.phone.get	70
acct.phone.set	72
acct.report.scheduled.delete	73
acct.report.scheduled.list	75
acct.search	78
acct.status.set	80

subacct.new	82
Account Addresses	85
acct.addr.city.get	86
acct.addr.city.set	87
acct.addr.country.get	88
acct.addr.country.set	90
acct.addr.postalcode.get	91
acct.addr.postalcode.set	92
acct.addr.state.get	94
acct.addr.state.set	95
acct.addr.street1.get	97
acct.addr.street1.set	98
acct.addr.street2.get	99
acct.addr.street2.set	100
User Settings	102
user.del	102
user.email.set	104
user.list	105
user.new	107
user.passwd.set	108
user.permissions.get	110
user.permissions.set	111
Ad Groups	113
group.list	114
group.new	115
group.search	117
group.user.permission.add	119
grp.descr.get	120
grp.descr.set	122
grp.name.get	123
grp.name.set	124
Ad Campaigns	126
ad.askexten.get	127
ad.askexten.set	128
ad.busr_hrs.get	129
ad.busr_hrs.set	132
ad.configure	135
ad.custom.define	143
ad.custom.get	146
ad.custom.set	149
ad.deactivate	151
ad.descr.get	153
ad.descr.set	154
ad.dispo.del	155

ad.dispo.list	157
ad.dispo.new	158
ad.dynamic.new	160
ad.good_user.set	163
ad.list.all	164
ad.list	169
ad.missed_user.set	171
ad.name.get	172
ad.name.set	174
ad.new	175
ad.settings.copy	177
ad.settings.get	178
ad.timezone.set	182
ad.vmail_user.set	184
timezone.list	185
Call Forwarding	186
ad.forw.assignto.set	187
ad.forw.del	189
ad.forw.exten.set	190
ad.forw.get	192
ad.forw.hrsmode.set	194
ad.forw.list	196
ad.forw.matchrule.set	199
ad.forw.new	200
ad.forw.status.set	202
ad.forw.step.set	204
ad.forw.timeout.set	205
ad.forw.weight.set	207
ad.forwtout.get	208
ad.forwtout.set	210
Call Recordings, Settings, and Search	211
ad.recordcall.get	212
ad.recordcall.set	213
ad.usevmail.get	214
ad.usevmail.set	215
call.audio	217
call.audio.url	219
call.custom.define	221
call.custom.set	225
call.get	227
call.set	242
Response Syntax: Basic Response	253
Response Syntax: Extended Response	257
Include Call DNA Response	257

Include Call Transcription Keywords Response	257
Include Call DNA	261
Include Spotted Keywords	262
Include High Intent	262
Include dni vars	264
call.transcript.get	267
Phone Numbers	269
number.avail	271
number.avail.npa	272
number.list	274
number.lookup.npanxx	277
number.order	279
number.order.list	280
number.order.npa	283
number.order.rc	285
number.provision.exact	286
number.provision.match_npanxx	288
number.provision.nearby	290
number.provision.npa	293
number.provision.rc	294
number.return	296
number.search.local.by.distance	298
number.search.npa	299
number.search.npa.list	301
number.search.npanxx.associated.list	303
number.search.npanxx.list	304
number.search.prefix	306
number.search.rc	309
number.search.state	311
Playfiles	313
Playfile Methods at the Account Level	314
Playfile Methods at the Ad Campaign Level	314
ad.play.file.new	315
ad.play.file.list	317
ad.play.afterhr_exten.set	319
ad.play.afterhr_exten.unset	320
ad.play.afterhrs.set	322
ad.play.afterhrs.unset	323
ad.play.busnhr_exten.set	325
ad.play.busnhr_exten.unset	326
ad.play.set	328
ad.play.unset	329

ad.play.vmail.set	331
ad.play.vmail.unset	332
ad.play.whisper.set	334
ad.play.whisper.unset	335
Postbacks	336
ad.call_postback.get	337
ad.call_postback.set	338
ad.precall_postback.get	339
ad.precall_postback.set	341
ad.calldna_postback.get	342
ad.calldna_postback.set	343
ad.callmodel_postback.get	345
ad.callmodel_postback.set	346
Rewrite Rules and JavaScript Rewrites	347
js.rewrite.list	347
js.rewrite.new	349

CHAPTER 6 Glossary

CHAPTER 1

About the Call Analytics JSON-RPC API

The Call Analytics API is based on the industry standard JSON-RPC, which is a remote procedure call protocol encoded in JSON. The Call Analytics JSON-RPC API provides methods to help you manage your accounts, users, and ad campaigns.

The Call Analytics API is ideal for Call Analytics platform users that require additional customization and functionality. If you are not familiar with Call Analytics UI, to understand the reference documentation on each method, you will need a grounding in Call Analytics concepts and terminology. See [Call Analytics Key Concepts and Terminology](#) for a primer.

Overview

Business Automation

The JSON-RPC API makes it easier to automate frequent activities.

- » Automate frequent reports and integrate Call Analytics data with an existing reporting infrastructure.
- » Automate account management tasks such as client account creation and number provisioning.

Streamlined Account Management

The JSON-RPC API provides functionality for managing multiple client accounts and for account related tasks.

- » Create setup defaults for managing ad campaigns.
- » Control customers' account access and branding experience with customized user interfaces and queries.

Flexible Reporting

The JSON-RPC API implements several flexible search procedures.

- » Get lists of account information, including ad campaigns and call logs, and limit results according to custom reporting needs.
- » Use custom queries to integrate Call Analytics data with your existing reporting infrastructure.

Requirements

JSON-RPC Support

You may need additional libraries if your programming language does not support JSON-RPC by default. See [Using PHP to Send a Request](#) and [Using Python to Send a Request](#) for information on the HTTP clients used in the example code for PHP and Python.

Call Analytics API Server Limitations

The Call Analytics API server limits each account to 10 concurrent requests. There is no limit on how often the server is accessed, as long as the total number of concurrent requests does not exceed 10. When you exceed your concurrent request limit, new method calls will immediately return this fault: *Connection Rejected - You Have Exceeded Your Connection Limit*.

API User Account and Permissions

While a special account is not required to use the API, we recommend creating a user account solely for API usage because using an API-specific user account will help in troubleshooting.

User access restrictions are enforced in the API as they are in the user interface, which means that the permissions required for the API user vary by the method. For example, to create a new ad campaign in the UI, a user must have **Account Admin** and **Allowed to Add Campaign** permissions in the account. If the API user does not have these permissions and calls **ad.new** to create a new ad campaign, the method will fault because the API user does not have the appropriate permissions.

Following are some requirements and guidelines for creating the API user:

- » You must log on to the Call Analytics user interface using a URL with the marchex.io domain name. Creating a user while logged on to the UI with a URL that contains a customized domain name is not supported.
- » To ensure the API user has the required permissions for every method, you may want to create the user in the parent account with all of the available permissions:
 - » Account Admin
 - » Client Admin
 - » Allowed to Add Campaign.
 - If you prefer, you can create different user accounts with different levels of permissions for the API just as you would do for the UI, but with this approach, you risk running into a permissions error.
- » Use a real email address in case you need to be contacted about a change or issue with the API.
- » If you create a new API user, make sure to log on to the Call Analytics UI once to change the password. The API user will be refused access until the password has been changed. Depending on the method, you may get back an HTTP 200 status code, but you will see a message like the following: "password must be reset to continue"

Limitations of the Call Analytics API

You can do most activities using the API that you can do using the web-based interface. However, the API does not enable some features:

- » The API cannot determine whether the call from one NPA-NXX combination to another is a local call, unless the distance between them is zero miles.
- » The API does not provide methods to remove account custom fields, ad custom fields, or call custom fields.

**This page intentionally left blank to ensure new
chapters start on right (odd number) pages.**

CHAPTER 2

Call Analytics Key Concepts and Terminology

Every software application introduces terminology and concepts to users in a UI and/or API. Call Analytics uses terms that may be familiar to you, such as *group* or *campaign*, but the meaning in Call Analytics may not be quite what you expect. In many ways, the Call Analytics API mirrors the UI. Just about every configuration that you can do in the UI you can also do in the API. If you are unfamiliar with the Call Analytics UI, you need to understand the terminology and concepts described in this section to understand the documentation for each method in the [API Reference](#).

Terminology Quick Reference

The terms in this section are a subset of the terms in the [Glossary](#). These are brief definitions that will give you a grounding for the more detailed explanations in the sections that follow.

Term	Definition
account	A collection of users and ad campaigns in ad groups . A top-level or parent account can contain client accounts, which may also be referred to as child accounts or sub-accounts.
ad, ad campaign, or campaign	An item in an account that tracks call activity. Every call that comes in on a call tracking number is associated with an ad campaign.
ad group	A collection of ad campaigns and users. An ad campaign can be a member of only one ad group, but users can be members of many ad groups. Users who are members of an ad group can see calls to the group's ad campaigns in the call log.
Call DNA	Call DNA graphically displays conversations of phone calls to your ad campaigns in the call log in the Call Analytics UI. Additionally, Call DNA analyzes the patterns in the conversation and classifies the result of the phone call. Whether or not you record phone calls, you can use the Call DNA visualization to see at a glance the outcome of a call.

Term	Definition
call log	A log that is surfaced as a page in the Call Analytics user interface that provides information about calls such as caller ID, ad campaign, time, duration, status, tags assigned, listened to, etc. Members of the ad campaign's ad group can click a call to view more information, see the Call DNA visualization of the call, or listen to the call if it was recorded.
call tracking number	A call tracking number (also referred to as an incoming number) is a phone number used to track calls to Call Analytics ad campaigns. A single call tracking number can be assigned to an ad campaign, or call tracking numbers can be dynamically assigned from a pool of numbers for use in dynamic tracking ad campaigns.
Dynamic Tracking	Dynamic tracking, also referred to as Keyword Level Tracking (KWLTL), is a Call Analytics feature that enables you to understand what keyword or session drove a specific call to your business or call center. With dynamic tracking, your campaigns use a pool of call tracking numbers. At a high level, when a visitor reaches your website, a JavaScript snippet runs that replaces your business phone number with a phone number from the dynamic pool. The phone number is leased to the visitor for a set period of time. If the visitor calls the phone number within the lease period, Call Analytics can tie the call data back to the website visit.
playfile	Playfiles play a greeting to callers to an ad campaign. Playfiles customize the caller experience and provide callers with details about the business or service they have reached.
postback	A one-time, one-way internet connection that notifies a server of an event, such as a call starting, a call ending, or submission of a web-based form.
rewrite	Rewrites change a website to show your ad campaign's call tracking number (or a tracking number from a dynamic number pool) based on how the visitor reached the website. A rewrite can replace a phone number on a website for every visitor, or only for visitors from a particular referring website or a particular paid ad.
rewrite rule	A rewrite contains one or more rewrite rules that specify which phone numbers on a website should be replaced with a call tracking number and which phone numbers should be excluded from the rewrite.
static campaign	Refers to a Call Analytics ad campaign that is configured with a single call tracking number. This type of campaign is also referred to as a one-to-one campaign because it has one call tracking number, in contrast to dynamic tracking campaigns .
user	A member of an account. A user is identified in the Call Analytics platform by their email address. Users can be assigned permissions that govern what functionality they have access to in the Call Analytics UI and API. Users created in a parent account can be granted access to the parent's child accounts. Users can be members of one or more ad groups that are in the account that they were created in.

Key Concepts

This section expands on the terminology introduced in ["Terminology Quick Reference" on page 12](#) and provides links to the descriptions of related methods. Additional information about the concepts described here can be found in the *Call Analytics User Guide*.

Accounts

An account is a collection of [users](#) and [ads](#) in [ad groups](#). Any other object that you can create in Call Analytics, (rewrites, proxies, playfiles, etc.) are associated with an account. A top-level or parent account can contain client accounts, which may also be referred to as child accounts or sub-accounts. Some example uses of parent and child accounts are:

- » An advertising agency may have a parent account, and each of the agency's advertisers may be created in separate child accounts.
- » A large national company may have a parent account, and regional offices may be represented in Call Analytics as separate child accounts.

Because an account is the primary grouping mechanism in Call Analytics, many methods require an account ID as an input parameter. A few older methods may use the account that the API user is a member of. (See ["API User Account and Permissions" on page 9](#) for more information.)

Account-related methods are described in the following sections:

- » ["Accounts" on page 37](#)
- » ["Account Addresses" on page 85](#)

Ad Campaigns

Ad campaigns (also referred to as ads or campaigns) are the central objects in Call Analytics. You create and configure a Call Analytics ad campaign to track your advertising campaign—whether it's for traditional media (radio, TV, newspaper, etc.) or for a paid ad in a search campaign. Ad campaigns are associated with one or more [call tracking numbers](#). For traditional media, you place the call tracking number in your ad instead of your business phone number. For search or display paid ads, you create and configure a Call Analytics [rewrite](#), and add a JavaScript snippet to your landing page to replace your business phone number with a call tracking number.

There are three types of ad campaigns in Call Analytics:

- » **Static:** Refers to a Call Analytics ad campaign that is configured with a single call tracking number. This type of campaign is also referred to as a one-to-one campaign because it has one call tracking number. To create static ad campaigns, you can use ["ad.new" on page 175](#) or ["ad.configure" on page 135](#).
- » **Dynamic:** With dynamic tracking, your campaigns use a pool of call tracking numbers. At a high level, when a visitor reaches your website, a JavaScript snippet is run that replaces your business phone number with a phone number from the dynamic pool. The phone number is leased to the visitor for a set period of time. If the visitor calls the phone number within the lease period, Call Analytics can tie the call data back to the website visit.

To create a dynamic tracking ad campaign, you use ["ad.dynamic.new" on page 160](#).

Ad campaigns have many settings. The input parameters in the ["ad.configure" on page 135](#) method mirror the Campaign Settings in the UI, as do many of the other ad campaign-related methods.

After the ad campaign has been created, you can configure them by using [ad.configure](#) or any of the other ad campaign-related methods described in the following sections:

- » ["Ad Campaigns" on page 126](#)
- » [***"Ad Groups" on page 113](#)
- » ["Call Forwarding" on page 186](#)
- » [**"Call Recordings, Settings, and Search" on page 211](#)
- » ["Playfiles" on page 313](#)
- » ["Postbacks" on page 336](#)

*There is some overlap in the groupings of the methods in the documentation. Only the methods that begin with "ad" are used to manage ad campaigns.

**When you create an ad campaign, you are required to supply the ID of an ad group. You may need to use one or more of the ad group methods to either find an existing group ID, or to create a new group.

Ad Groups

Ad groups (also called groups) are a collection of users and ad campaigns in the same account. Ad groups allow you to organize your ad campaigns for reporting and management purposes. They also provide a convenient way to track calls to different categories. For example, there can be a group for customer support and another group for sales. These two ad groups can be monitored and reported on separately. An ad campaign can be a member of only one ad group, but users can be members of many ad groups.

Users who are members of an ad group can see calls to the group's ad campaigns in the call log. Call Analytics can send an email to an ad group member for each good call, missed call, and voicemail to an ad campaign. You can filter reports by ad group.

- » Every account has a normal ad group named `default`. If you don't need group features, just put your users and ads in this ad group.
- » Each ad group resides in just one account. Ad groups are not shared across accounts. The parent account has its own ad groups, as does each client account.

Ad group-related methods are described in the following section:

- » ["Ad Groups" on page 113](#)

Call Log

Every call that comes in on a call tracking number is logged in the Call Analytics call log. The call log that is surfaced as a page in the Call Analytics user interface provides information about calls such as caller ID, ad campaign, time, duration, status, tags assigned, listened to, etc. Members of the ad campaign's ad group can click a call to view more information, see the Call DNA visualization of the call, or listen to the call if it was recorded.

You use the ["Response Syntax: Basic Response" on page 253](#) method to search the call log to find calls that you want more information on. The response to **call.search** contains detailed information about each call, such as date, time, duration, status, Call DNA outcome, etc.

The following section describes the methods available to search the call log and to get and set information about individual calls.

- » ["Call Recordings, Settings, and Search" on page 211](#)

Call Tracking Numbers

At a high level, when customers call the call tracking number that is associated with a Call Analytics campaign, the call is routed to a Call Analytics data center. Call Analytics then finds the campaign associated with the call tracking number, and forwards the call to a phone number that has been configured on the campaign (which is referred to as the forward to number).

There are two pools of call tracking numbers:

- » **Dynamic pool:** To obtain call tracking numbers for dynamic ad campaigns, you must contact Client Services. To enable dynamic tracking, Client Services will need the name of the account and/or the account ID, as well as the desired pool size and lease period. Client Services can recommend a pool size and lease period if you provide website traffic or click-through data. There are no API methods that manipulate the dynamic pool of call tracking numbers.
- » **Number pool associated with a parent account:** These are the numbers that you use for static ad campaigns. You can use the methods described in the ["Phone Numbers" on page 269](#) section to manage the call tracking numbers in the parent account's number pool. Many of the methods require that you first contact Client Services to enable a feature called *Real Time Number Provisioning*. This setting allows you to provision a number in real time rather than waiting for a number order to be completed by Client Services.

Rewrites

Rewrites change a website to show your ad campaign's call tracking number based on how the visitor reached the website. A rewrite can replace a phone number on a website for every visitor, or only for visitors from a particular referring website or a particular paid ad. You can also show a call tracking number based on which search phrase a visitor entered into the search engine. Call Analytics provides the following kinds of rewrites:

JavaScript Rewrites

JavaScript rewrites (also referred to as static rewrites because they are associated with static ad campaigns) can be used for both paid and organic traffic. When a JavaScript rewrite triggers, it replaces phone numbers on your landing page with a call tracking number.

Because you must add an account-specific JavaScript snippet to your landing page (and every page in the website that you want the rewrite to trigger on), you must have access to your website's HTML source code to use a JavaScript rewrite. The following section describes the methods available for JavaScript rewrites:

- » ["Rewrite Rules and JavaScript Rewrites" on page 347](#)

Dynamic Tracking Rewrites

Dynamic tracking rewrites (referred to as Keyword Tracking JavaScript Rewrites in the Call Analytics user interface) are part of the Dynamic Tracking feature. Dynamic Tracking enables you to understand what keyword or session drove a specific call to your business. When a dynamic tracking rewrites triggers, it replaces phone numbers on your landing page with a call tracking from a dynamic number pool.

Because dynamic tracking rewrites will trigger only when certain URL parameters are in the [landing page URL](#), and because some search engines do not pass on keyword data when a visitor clicks on an organic search result to reach your landing page, dynamic tracking rewrites should be used only with paid traffic.

Because you must add an account-specific JavaScript snippet to your landing page (and every page in the website that you want the rewrite to trigger on), you must have access to your website's HTML source code to use a dynamic tracking rewrite.

You create a dynamic tracking rewrite when you create the dynamic tracking ad campaign, using ["ad.dynamic.new"](#) on [page 160](#).

Users

A member of an account. A user is identified in the Call Analytics platform by their email address. Users can be assigned permissions that govern what functionality they have access to in the Call Analytics UI and API. Users can be members of one or more ad groups that are in the account that they were created in.

You can create users in a parent account or in client accounts. You can give parent account users permission to be members or administrators of all client accounts or only to the client accounts that you specify. You can grant parent account users the following permissions:

- » **Account Admin:** Gives the user Account Admin permission on the parent account.
- » **Client Admin:** Gives the user Account Admin permission on all client accounts.
- » **Client Viewer:** Allows the user to view all client accounts.
- » **Allowed to Add Campaign:** Lets the user create campaigns in the accounts that they have Admin permissions in. If the user also has the Account Admin permission selected, allows the user to create campaigns in the parent account. If the user also has the Client Admin permission selected, allows the user to create campaigns in all client accounts. If you want to limit which client accounts the user can create campaigns in, enable Allowed to Add Campaigns, and then select the client accounts that you want the user to be an Admin in.

Users created in client accounts can be granted only the Account Admin and the Allowed to Add Campaign permissions.

User-related methods are described in the following section:

- » ["User Settings" on page 102](#)
- » [*"Ad Groups" on page 113](#)

*You use the **group.user*** methods to add and remove users to groups and to set user permissions in a group.

CHAPTER 3

Getting Started

This section describes the information you need to get started using the Call Analytics JSON-RPC API. This section does not cover JSON or JSON-RPC basics. If you need a primer, following are a few of the reliable sources of information available online:

JSON

- » <http://www.tutorialspoint.com/json/index.htm>
- » http://www.w3schools.com/js/js_json_intro.asp
- » <https://en.wikipedia.org/wiki/JSON>

JSON-RPC:

- » <http://www.jsonrpc.org/>
- » <https://en.wikipedia.org/wiki/JSON-RPC>

JSON syntax checkers:

- » <http://jsonlint.com/>
- » <https://jsonformatter.curiousconcept.com/>

As with all JSON-RPC methods, you need to know the following to construct a request:

- » The endpoint, which is the URL where the Web service can be accessed by a client application.
- » The authentication method.
- » The expected headers.
- » The request syntax and parameters.

Endpoint

The Call Analytics JSON-RPC API can be accessed at the following URL:

`https://api.marchex.io/api/jsonrpc/1`

Basic Access Authentication

The Call Analytics JSON-RPC API uses the HTTP standard Basic Access Authentication, a simple authentication method where you provide a user name and password when making a request. To use the API, create an API user in the applicable Call Analytics account. When sending a request to a method, you pass in the email address and password of the API user. These are the same credentials that the API user would use to log on to the Call Analytics user interface.

Note The permissions required for an API request matches the permissions required to do the same task in the UI. See [API User Account and Permissions](#).

If you create a new user, the API user should first log on to the UI and change their password.

HTTP Header Fields

The Call Analytics JSON-RPC APIs use the following standard HTTP header fields:

- » "Content-Type": "application/json"
- » "Accept": "application/json"
- » "Authorization": "Basic <Base64 encoded username:password>"

Request Syntax

The end point, authentication method, and the HTTP headers are the same for all of the Call Analytics JSON-RPC methods. The request syntax varies by method depending on the request parameters that the method requires.

Following is the request syntax for the ["timezone.list" on page 185](#) method:

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "timezone.list",
  "params": [
  ]
}
```

The request syntax for all of the JSON-RPC methods follow a similar pattern. The following key-value pairs are the same for every method:

- » "jsonrpc": "2.0"
- » "id": 1

The "method" key takes the name of the method as a string value.

Following is the request syntax for the ["acct.list" on page 59](#) method:

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.list",
  "params": [
  ]
}
```

```
}  
}
```

As far as the request syntax goes, the only difference from **timezone.list** is that "acct.list" is the value for the "method" key. The main difference is that **timezone.list** does not require any special permissions of the API user, but **acct.list** requires the API user to have Account Admin permissions.

Request Parameters

The **timezone.list** and **acct.list** methods are the simplest methods to send requests to because they do not take any parameters, as you can see from the empty *params* array in the request syntax for the methods.

All requests include the *params* array, but the parameters vary by method, for example:

- » Most methods, such as [grp.name.set](#) require one or more positional parameters in the request. When a method takes positional parameters, all parameters are required in the request, and the order of the parameters must match the order given in the Request syntax in the method's documentation.
- » Some methods, such as [group.search](#) require an object containing key-value pairs in the request. When a method takes an object, depending on the method, some of the key-value pairs may be required in the request, and some may be optional.
- » Some methods, such as [ad.list.all](#) take positional parameters that include an object. Even when all the key-value pairs in the object are optional, as is the case with **ad.list.all**, you still must specify the object by including a pair of empty braces ({} in the request.

Method Names

The first part of a method name describes the Call Analytics "entity" that the method works on. For example:

- » Methods that begin with "acct" such as [acct.list](#) and [acct.name.get](#) operate on accounts. As you can probably surmise by the names, **acct.list** returns a list of accounts and **acct.name.get** returns that name of an account.
- » Methods that begin with "grp" or "group" such as [group.search](#) and [grp.name.get](#) operate on ad groups.
- » Methods that begin with "ad" such as [ad.configure](#) operate on ad campaigns.

The [JSON-RPC Reference](#) provides a list of the different method groupings.

Note Many of the methods perform their action (get, set, configure, etc.) on one entity (account, ad campaign, call, user, etc.). To perform the action on more than one entity, you have to send the request again. If a method can perform its action on more than one entity, it is clearly noted in the method's documentation. For example, [call.audio.url](#) takes an array of call IDs in the request.

IDs

Most of the methods require a system-generated ID for the Call Analytics entity that the method accesses. The IDs are strings and currently they contain 16 characters. Although the string length of an ID is unlikely to change, if you are building an application using the Call Analytics API, you probably shouldn't use a fixed-length array for the ID.

The first ID that you will need is the account ID of the account that the API user is a member of. Typically, this is a parent account. If you do not know the account ID for the API user, you can call "[acct.list](#)" on [page 59](#) to get it.

Once you have the account ID that you need, you can call other "list" or "search" methods to obtain other IDs. For example:

- » Accounts: [acct.search](#)
- » Ad campaigns: [ad.list](#) or [ad.list.all](#)
- » Ad groups: [group.list](#) or [group.search](#)
- » Calls: [call.search](#)
- » Playfiles: [ad.play.file.list](#)
- » Users: [user.list](#)

Send a POST Request via curl

For you Linux command line users, following is the complete request to **acct.list** using curl. You just need to modify the line that starts with **-u** and add the email address and password of the API user.

```
curl -d \
'{"jsonrpc": "2.0", "id": 1, "method": "acct.list", "params": []}' \
-H "Content-Type: application/json" \
-H "Accept: application/json" \
-X POST \
-u "your_alias@your_company.com:your_password" \
"https://api.marchex.io/api/jsonrpc/1"
```

Using Postman to Send a Request

If you are new to programming, or if you want to quickly test a method, you may want to use a Web REST client such as Postman for Google Chrome to send requests to the methods. Although this section provides steps for using Postman, this should not be construed as an endorsement of Postman over similar browser add-ons such as Poster for Firefox. Note that the Postman steps were written using Postman version 4.1.3 on Microsoft Windows. Although the user interface may be slightly different on other versions of Postman, the concepts are the same.

Task List

Use the following task list and the detailed procedures that follow to send a POST request to **"acct.list"** on page 59.

1. "Install Chrome and Postman" below.
2. "Set the end point" on the facing page.
3. "Set the authentication method and credentials" on the facing page.
4. "Set the HTTP headers" on the facing page.
5. "Post a request to acct.list" on the facing page.

Procedures

Install Chrome and Postman

1. If you don't already have Chrome, download it from: <https://www.google.com/intl/en/chrome/browser/desktop/index.html>
2. Start Chrome, and download Postman from: <https://chrome.google.com/webstore>
3. Start Postman: in Chrome, click the **Apps** bookmark and click **Postman**.

Although the following procedures provide steps on configuring Postman to send a POST request to the **acct.list** method, you may want to review the Postman documentation to see screen shots and for additional information, such as how to save a request to a file. The Postman documentation is available from: <http://www.getpostman.com/docs>

Set the end point

The Call Analytics JSON-RPC API is accessed at the following URL:

`https://api.marchex.io/api/jsonrpc/1`

1. Copy and paste the above URL in the **Enter Request URL** box.
2. Select **POST** from the list of HTTP verbs.

Set the authentication method and credentials

1. In the Authorization section, select **Basic Auth** from the Type list.
2. In the Username box, enter the email address of the API user.
3. In the Password box, enter the password of the API user.
4. Click the **Save helper data to request** check mark.
5. Click **Update Request**.

Set the HTTP headers

1. Click **Headers**.
2. Click in the **key** box in the first row and press the down arrow on your keyboard to display the list.
3. Select **Content-Type** from the list.
4. Click in the **value** box, and select **application/json** from the list.
5. Click in key box in the next row and select **Accept** from the list.
6. Click in the **value** box, and select **application/json** from the list.

Post a request to acct.list

Following is the request syntax for the "**acct.list**" on page 59 method:

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.list",
  "params": [
  ]
}
```

1. Click **Body**.
2. Click **raw**. The request text box displays.
3. Copy and paste the above **acct.list** request into the Postman request text box.
4. Click **Send**.

If the request is successful, the response will be an array of objects, each of which contains information about an account, similar to the following:

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "acct": "CA6ph0qob-ZxRgJP",
      "customid": "",
      "status": "active",
      "name": "Test account 1"
    },
    {
      "acct": "CA6ph0qob-ZxRgFA",
      "customid": "",
      "status": "active",
      "name": "Test account 2"
    },
    {
      "acct": "CA6ph0qob-ZxTYJP",
      "customid": "",
      "status": "active",
      "name": "Test account 3"
    },
    {
      "acct": "CA6ph0qob-ZYaaJR",
      "customid": "",
      "status": "active",
      "name": "Test account 4"
    }
  ]
}
```

If the request is unsuccessful, the response will contain the reason why. For example:

- » **Not Found:** This means there is a typo in the endpoint.
- » **Authorization required:** This means that the user credentials are not valid. You are either using the wrong user name and password, or there is a typo. If you don't spot the problem, try logging on to the Call Analytics user interface using the same credentials.
- » An object containing the message "**Permission denied; Not an Account Admin**": The API user does not have the Account Admin permission. You will need to get your Call Analytics administrator to grant the API user the Account Admin permission.

When constructing the body for other methods, you can copy/paste from either the Request Syntax or Example sections of the reference documentation for the method and then make the needed modifications.

Using Python to Send a Request

This section describes how to send a POST request to ["acct.list" on page 59](#) via Python. You will need the following:

- » Credentials for the API user. Note the following:
- » Make sure that you log on to the Call Analytics UI using the API user's credentials at least once to change the password. If you don't change the password for the API user, the request will fail.
- » Make sure that the API user has Account Admin permissions. If the API user is a member of a parent account and you want a list of sub-accounts, the API user must have Account Admin and either Client Admin or Client Viewer permissions.
- » A Python development environment. The example code was written using the 3.x version of the IDLE Python GUI on Windows, which is included in the Python installation from python.org.
- » An HTTP client. The example code uses the Python Requests package, which greatly simplifies making HTTP requests and processing the responses. The Requests package is not included in the standard Python installation. For information on downloading and installing the Requests package, see [Requests: HTTP for Humans](#).

The example code also uses the standard json package, which is included in most Python installations.



Note: If you download **acct-list.py** to Linux or Mac OS X, you will need to run `dos2unix` on the file to convert the line endings. If you don't have `dos2unix`, you can download it from sourceforge.net.

Downloading Python and pip

If you do not already have Python and/or pip (the tool for installing Python packages), the following sections point to some reliable sources of information online. If you are new to programming in Python, a good resource is: [Google's Python Class](#).

pip

pip is already installed if you're using Python 2.7.9 or later, or Python 3.4 or later from python.org. However, you may need to upgrade it. For more information, see [Do I need to install pip?](#)

Python on Windows

The example code was written using Python 3.x. If you are using Python 2.x, the sample code will still run, but the parentheses surrounding the parameters in the `print` statements, will print tuples, so you may want to remove the parentheses. Both Python 2.x and Python 3.x for Windows can be downloaded from python.org.

Python on Linux

For Linux users, Python should be included in your Linux distribution. With the addition of the shebang line (`#!/usr/bin/python`) at the top of the file, the example code ran fine on Ubuntu 12.04.5 with the following changes to the `acct-list.py` file that was copied from Windows:

- » `dos2unix acct-list.py`
- » Make the file executable, for example: `chmod +x acct-list.py`

Depending on the location of your Python installation, you may need to alter the path in the shebang line. To see the path to your Python installation, at the command line, enter:

```
which python
```

For more information, see [Installing Python on Linux](#).

Python on the Mac

Mac OS X 10.8 comes with Python 2.7. Because it was written in Windows, you will need to make the following changes to the `acct-list.py` file after you download it to the Mac:

- » `dos2unix acct-list.py`
- » Make the file executable, for example: `chmod +x acct-list.py`

For more information, see [Python Releases for Mac OS X](#) and [Using Python on a Macintosh](#).

The Complete Example Code

This section provides the complete example code which you can copy and paste into a file, but if you prefer, you can download [python-examples.zip](#), which contains the `acct-list.py` file.

If you are new to Python, you need to know that Python enforces proper indentation to make code more readable. Python will not execute a file if it is not indented properly. If you copy and paste the following example code into a file, you may need to adjust the indentation.

To run the example:

- » Change *user* and *password* to the credentials of the API user.

```
#!/usr/bin/python

import json
import requests # Available from: http://docs.python-requests.org/en/master

def main():
    # Change to API user's email address
    user = "api_user@your_company.com"
    # Change to API user's password
    password = "password"
    request_body = {
        "jsonrpc": "2.0",
        "id": 1,
        "method": "acct.list",
        "params": []
    }

    response = send_request(user, password, request_body)

    # HTTP status code of 200 indicates a successful response.
    if response.status_code == 200:
        account_list = response.json()['result']
        for account in account_list:
            if account['status'] == 'active':
                print (account['acct'], " Name: ", account['name'])
    else:
        #Got an HTTP error response.
        print (response.status_code, response.text)

def send_request(user, password, request_body):
```

```

    endpoint = 'https://api.marchex.io/api/jsonrpc/1'
    headers = {'Content-type': 'application/json', 'Accept': 'application/json'}
    response = requests.post(endpoint, json=request_body, headers=headers, auth=(user-
, password))
    return response

if __name__ == '__main__':
    main()

```

A Close Look at the Example Code

This section provides a detailed explanation of the example code.

The Shebang Line

```
#!/usr/bin/python
```

The shebang line is needed only if you are running the code on Linux or Mac OS X. For Windows, you can either delete this line, or leave it since it starts with the Python comment character.

The Packages

```
import json
import requests
```

The json package is included with most (if not all) Python installations. If you don't already have it, you can install the Requests package from [Requests: HTTP for Humans](#).

Set the Credentials

The only modification that you need to make to the example code is to set the *user* and *password* variables to the email address and password of the API user.

```

# Change to API user's email address
user = "your_alias@your_company.com"
# Change to API user's password
password = "your_password"

```

The Request Body

The *request_body* variable is a Python dictionary. If you look at the request syntax in the reference documentation for ["acct.list" on page 59](#), you will see that the syntax is exactly the same as the *request_body* variable. When constructing the *request_body* for other methods, you can copy/paste from either the Request Syntax or Example sections of the reference documentation for the method and then make the needed modifications.

```

request_body = {
    "jsonrpc": "2.0",
    "id": 1,
    "method": "acct.list",
    "params": []
}

```

Encapsulating the POST Request

The details of the request are encapsulated in a function called **send_request**, which takes the credentials of the API user and the *request_body* as parameters. Here's the function call:

```
response = send_request(user, password, request_body)
```

By encapsulating the details of the request in a function, you can easily call other methods by simply changing the *request_body* variable. If you will always be using the same API user credentials, you could move the *user* and *password* variables to within the **send_request** function rather than passing them as parameters. The example code was set up with the *user* and *password* variables near the top of the file simply to alert you that these variables need to be changed to the API user's credentials. Here's the function:

```
def send_request(user, password, request_body):
    endpoint = 'https://api.marchex.io/api/jsonrpc/1'
    headers = {'Content-type': 'application/json', 'Accept': 'application/json'}
    response = requests.post(endpoint, json=request_body, headers=headers, auth=(user-
,password))
    return response
```

By using the Requests package, the process of sending the request is simplified to:

- » Setting the endpoint (the URL to the JSON-RPC service).
- » Setting the headers.
- » Specifying the authentication method and credentials.
- » Setting the request body.
- » Calling the **requests.post** method to make the request.

For more information, see [Requests: HTTP for Humans](#).

Processing the Response

When a request is successful, the Call Analytics JSON-RPC methods return the HTTP status code of 200. If there was an error, such as bad credentials or insufficient permissions, the API returns an HTTP error code. See "[HTTP Status Codes](#)" on page 37 for more information.

If the request is successful, the response is an object (that is, a Python dictionary), and it always contains a key called *result*. The *result* varies depending on the method. For **acct.list**, the *result* is an array of objects, each of which contains information about an account. In Python terminology, the *result* is a list of dictionaries.

The code checks to see if the account is active, and if it is, it prints out the account ID and the account name.

```
# HTTP status code of 200 indicates a successful response.
if response.status_code == 200:
    account_list = response.json()['result']
    for account in account_list:
        if account['status'] == 'active':
            print (account['acct'], " Name: ", account['name'])
else:
    #Got an HTTP error response.
    print (response.status_code, response.text)
```

The reference documentation for each method describes the response syntax in the *result* object. If you want to see the complete response for a method, you can simply do:

```
print (response.json())
```

To see just the *result* object:

```
print (response.json()['result'])
```

Using PHP to Send a Request

This section describes how to send a POST request to ["acct.list" on page 59](#) via PHP. You will need the following:

- » Credentials for the API user. Note the following:
- » Make sure that you log on to the Call Analytics UI using the API user's credentials at least once to change the password. If you don't change the password for the API user, the request will fail.
- » Make sure that the API user has Account Admin permissions. If the API user is a member of a parent account and you want a list of sub-accounts, the API user must have Account Admin and either Client Admin or Client Viewer permissions.
- » A web server with a PHP engine. The [WAMP SERVER](#) Windows web development environment was used to run the sample, but the sample should work on any web server with a PHP engine.
- » An HTTP client for PHP. There are many HTTP clients available for download for free. The sample code uses the Requests for PHP library. To install the Requests for PHP library:
 1. Go to [Requests for PHP](#), and click the **Download Now** button to download a .zip file.
 2. Copy the .zip file to your preferred directory, and extract it. For example: C:\wamp64\www\Requests-1.7.0\

The example code also uses the standard PHP functions `json_encode` and `json_decode`, which are included with PHP 5.2.0 and later. You can download the example code from [php-examples.zip](#).



Note: If you are using the Requests for PHP library, you will need to increase the timeout value in `Request.php` from the default of 10 seconds to at least 30.

Other HTTP Clients

Although the example code uses Requests for PHP, this should not be construed as an endorsement. There are many other HTTP clients available, for example:

- » [HTTPFUL](#)
- » [Guzzle](#)
- » [simple-http-client](#)



Note: If you use another HTTP client, you will need to make a few modifications to the example code.

The Complete Example Code

To run the example

- » Change *\$request_lib_path* to the path where you have installed the Requests for PHP package. Note that the path must end with the Request.php file name. For example:

```
$request_lib_path = 'C:\wamp64\www\Requests-1.7.0\library\Requests.php';
```

- » Change *\$user* and *\$password* to the credentials of the API user.

```
<!DOCTYPE HTML>
<html>
<head>
<title>acct.list</title>
</head>
<body>
<?php

// Change the following 3 variables.
$request_lib_path = 'C:\path_to\requests\library\Requests.php'; // Change to Requests lib-
rary path.
$user = 'api_user@some_company.com'; // Change to API user's email address
$password = 'password'; // Change to API user's password

// Include Requests.
include($request_lib_path);

// Make sure Requests can load internal classes.
Requests::register_autoloader();

// Prepare the data required by the method.
$method = 'acct.list';
$params = array();

// Send the request.
$response = send_request($user, $password, $method, $params);

// Process the response.
process_response($response);

function send_request($user, $password, $method, $params) {
    // The url, headers, and authentication are the same for all methods.
    $url = 'https://api.marchex.io/api/jsonrpc/1';
    $headers = array('Content-Type' => 'application/json');
    $options = array('auth' => new Requests_Auth_Basic(array($user, $password)));

    $data = array('jsonrpc' => '2.0', 'id' => 1, 'method' => $method, 'params' => $params);

    // POST to the API.
    $response = Requests::post($url, $headers, json_encode($data), $options);
    //var_dump($response->body);
    return $response;
}

function process_response($response) {
```

```

$response_array = json_decode($response->body, true);
$account_list = $response_array['result'];

// Print active accounts.
echo "<pre>";
foreach($account_list as $account) { //foreach account in $account_list
    if ($account['status'] == "active") {
        //var_dump($account);
        echo $account['acct'], "   Name:   ", $account['name'], "<br>";
    }
}
echo "</pre>";
}

?>
</body>
</html>

```

A Close Look at the Example Code

This section provides a detailed explanation of the example code.

Change the Values of Three Variables

To run the example, you need to change the following variables. This is the only modification that you need to run the example.

```

$request_lib_path = 'C:\path_to\requests\library\Requests.php'; // Change to Requests library path.
$user = 'api_user@company.com'; // Change to API user's email address.
$password = 'password'; // Change to API user's password.

```

Include and Register the Request for PHP Library

The include statement uses the *\$request_lib_path* variable that you specified above.

```

// Include Requests
include($request_lib_path);
// Make sure Requests can load internal classes
Requests::register_autoloader();

```

Prepare the Data Required by the Method

Every method in the JSON-RPC API requires that you specify the method name and the parameters in an array. Even though **acct.list** does not take any parameters, you still must create an empty array for the request.

```

$method = 'acct.list';
$params = array();

```

Encapsulating the POST Request

The details of the request are encapsulated in a function called **send_request**, which takes the credentials of the API user, *\$method*, and *\$params* as parameters. Here's the function call:

```
$response = send_request($user, $password, $method, $params);
```

By encapsulating the details of the request in a function, you can easily call other methods by simply changing the values assigned to the *\$method* and *\$params* variables. If you will always be using the same API user credentials, you could move the *\$user* and *\$password* variables to within the **send_request** function rather than passing them as parameters. The example code was set up with the *\$user* and *\$password* variables near the top of the file simply to alert you that these variables need to be changed to the API user's credentials. In the *\$data* associative array, the first two key/value pairs are the same for each method. Here's the function:

```
function send_request($user, $password, $method, $params) {
    // The url, headers, and authentication are the same for all methods.
    $url = 'https://api.marchex.io/api/jsonrpc/1';
    $headers = array('Content-Type' => 'application/json');
    $options = array('auth' => new Requests_Auth_Basic(array($user, $password)));

    $data = array('jsonrpc' => '2.0', 'id' => 1, 'method' => $method, 'params' => $params);

    // POST to the API.
    $response = Requests::post($url, $headers, json_encode($data), $options);
    //var_dump($response->body);
    return $response;
}
```

By using the Requests for PHP library, the process of sending the request is simplified to:

- » Setting the endpoint (the URL to the JSON-RPC service).
- » Setting the headers.
- » Specifying the authentication method and credentials.
- » Setting the data.
- » Calling the **Requests::post** method to make the request.

For more information, see [Requests for PHP](#).

Processing the Response

When a request is successful, the Call Analytics JSON-RPC methods return the HTTP status code of 200. If there was an error, such as bad credentials or insufficient permissions, the API returns an HTTP error code. See "[HTTP Status Codes](#)" on [page 37](#) for more information.

If the request is successful, the response is an object (that is, a PHP associative array), and it always contains a key called *result*. The *result* varies depending on the method. For **acct.list**, the *result* is an array of objects, each of which contains information about an account. In PHP terminology, the *result* is an array of associative arrays.

Just to be tidy, the processing of the response is encapsulated in a function. Here's the call:

```
process_response($response);
```

The example simply outputs the account ID and name for all the active accounts. Here's the function:

```

function process_response($response) {
    $response_array = json_decode($response->body, true);
    $account_list = $response_array['result'];

    // Print active accounts.
    echo "<pre>";
    foreach($account_list as $account) { //foreach account in $account_list
        if ($account['status'] == "active") {
            //var_dump($account);
            echo $account['acct'], "   Name:  ", $account['name'], "<br>";
        }
    }
    echo "</pre>";
}

```


**This page intentionally left blank to ensure new
chapters start on right (odd number) pages.**

CHAPTER 4

Example Code for JSON-RPC API

The following code examples are intended as a hands-on introduction to the Call Analytics JSON-RPC API. The examples are provided as reference material and are not intended for live deployment. Consult your language reference regarding language specific issues such as error handling and best practices for security and deployment.

Python Example Code

- » The Python example code is available here: [python-examples.zip](#)

The following files are included in the Python examples:

- » **acct-list.py**: A simple example to get started using the JSON-RPC API. For a detailed explanation of this example code, see [Using Python to Send a Request](#).
- » **timezone.list.py**: Another simple example to get started using the JSON-RPC API. You may need to make a request method to [timezone.list](#) to make sure you have the correct string for the *time_zone* parameter when creating an ad campaign via [ad.configure](#).
- » **call-search-plus.py**: This example show how to make a request to [call.search](#), add the call IDs in the response to a list, and then pass the list of call IDs in a request to [call.audio.url](#).
- » **ad-configure.py**: This example shows how to make a request to [ad.configure](#) to either create a new ad campaign or modify an existing campaign.
- » **ad-configure-csv.py**: This example shows how to read from .csv files to get the campaign data to send to the [ad.configure](#) method.
- » **ad-play-file-new.py**: This example shows how to read a .wav file and convert it to a Base64-encoded text string to send to [ad.play.file.new](#). The playfile ID that is returned by [ad.-play.file.new](#) is sent to [ad.play.set](#) to set the playfile as the business hours greeting.

PHP Example Code

- » The PHP example code is available here: [php-examples.zip](#)

The following files are included in the PHP examples:

- » **acct-list.php**: A simple example to get started using the JSON-RPC API. For a detailed explanation of this example code, see [Using PHP to Send a Request](#).
- » **timezone.list.php**: Another simple example to get started using the JSON-RPC API. You may need to make a request method to [timezone.list](#) to make sure you have the correct string for the *time_zone* parameter when creating an ad campaign via [ad.configure](#).
- » **call-search-plus.php**: This example show how to make a request to [call.search](#), add the call IDs in the response to a list, and then pass the list of call IDs in a request to [call.audio.url](#).
- » **ad-configure.php**: This example shows how to make a request to [ad.configure](#) to either create a new ad campaign or modify an existing campaign.
- » **ad-configure-csv.php**: This example shows how to read from .csv files to get the campaign data to send to the **ad.configure** method.
- » **ad-play-file-new.php**: This example shows how to read a .wav file and convert it to a Base64-encoded string to send to [ad.play.file.new](#). The playfile ID that is returned by **ad.play.file.new** is sent to [ad.play.set](#) to set the playfile as the business hours greeting.

CHAPTER 5

JSON-RPC Reference

The JSON-RPC Reference provides detailed information on each method. The methods are grouped into the following categories:

Accounts: Describes the methods available to manage accounts.

Account Addresses: Describes the methods you use to get and set account addresses.

User Settings: Describes the methods available to manage users.

Ad Groups: Describes the methods available to manage ad groups. After adding users, review the methods in this section to set group permissions on the users.

Ad Campaigns: Describes the methods you use to get and set ad campaign settings.

Call Forwarding: Describes the methods to get and set call forwarding settings on ad campaigns. After adding campaigns and configuring settings on them, review the methods in this section to add the forwarding rules for the campaigns.

Call Recordings, Settings, and Search: Describes the methods that deal with call recordings, call-related settings on ad campaigns, and searching for calls in the call log.

Phone Numbers: Describes the methods available to search, order, and provision phone numbers.

Playfiles: Describes the methods that allow you to control the greetings that are played to callers.

Postbacks: Describes the methods allow you to get and set postback URLs on ad campaigns.

Rewrite Rules and JavaScript Rewrites: Describes the methods that allow you to manage rewrite rules and JavaScript rewrites.

Proxy and Proxy Rewrites: Describes the methods available to manage proxies and proxy rewrites.

Images: Describes the methods available to control the images used for proxy image replacement.

Wind Down: Describes the methods that deal with wind down settings. Wind down is an optional period near the end of an ad campaign where the caller hears a wind down playfile or a new number to call in the future.

Click-to-Call: Describes the methods available for click-to-call ad campaigns.

HTTP Status Codes

Operations that you perform with the JSON-RPC API return an HTTP status code. The following table shows the most common status codes and descriptions. In many cases, a more specific error message is returned in the JSON response rather than the default message.

Code	Default Message	Description
200	Success	The operation completed successfully.
400	Invalid Request	The request was missing the required parameter(s), or the request could not be understood by the server due to malformed syntax. The client should not repeat the request without modifications.
401	Unauthorized	The API user's credentials are invalid.
402	Payment Required	You may see this status code when the request should return information about a feature that is not enabled. For example, <code>call.search</code> returns this status code if the request has <code>include_dna</code> set to true, but Call DNA is disabled on the account.
403	Permission Denied	The API user does not have the required permissions.
404	Resource Not Found	A resource specified in the request (such as an account ID, campaign ID, etc.) could not be found.
409	Conflict	The request could not be completed due to a conflict with the current state of the resource. You may see this, for example, when a request for a number with a specific area code is unavailable.
410	Resource Unavailable	The requested resource is no longer available.
415	Unsupported Media Type	The request failed because the entity of the request is in a format that is not supported. This status code is returned, for example, by the <code>ad.play.<playfile_type>.set</code> methods if the specified playfile is not in MP3 or WAV format.
500	A server error occurred	The operation failed because of a server error.

Accounts

This section describes the methods available to manage accounts.

Method	Description
<code>acct.category.get</code>	Gets the category of the specified account.
<code>acct.category.set</code>	Sets the category of the specified account.

Method	Description
<u>acct.custom.define</u>	Defines a custom field for the specified account.
<u>acct.custom.get</u>	Gets the value of the specified custom field for the specified account.
<u>acct.custom.set</u>	Sets the value of the specified custom field for the specified account.
<u>acct.customid.get</u>	Gets the custom reference ID for the specified account.
<u>acct.customid.set</u>	Sets the custom reference ID for the specified account.
<u>acct.keyword_spotting.word_list.get</u>	Gets the list of call transcription keywords that have been set for each speaker (the agent and/or caller) on the specified account.
<u>acct.keyword_spotting.word_list.set</u>	Sets the specified list of call transcription keywords for each speaker (the agent and/or caller) on the specified account.
<u>acct.list</u>	Gets a list containing account information for the parent account that the API user was created in and all client accounts.
<u>acct.membership.list</u>	Gets a list containing account membership information for the specified user.
<u>acct.membership.set</u>	Sets the account membership information for the specified user and account.
<u>acct.membership.unset</u>	Unsets account membership information for the specified account and user.
<u>acct.name.get</u>	Gets the name of the specified account.
<u>acct.name.set</u>	Sets the name of the specified account.
<u>acct.numlookup.set</u>	Sets whether reverse lookup for incoming calls is enabled for the specified account
<u>acct.phone.get</u>	Gets the telephone number for the contact information of the specified account.
<u>acct.phone.set</u>	Sets the telephone number for the contact information of the specified account.
<u>acct.report.scheduled.delete</u>	Deletes the specified scheduled report.
<u>acct.report.scheduled.list</u>	Gets a list containing information about scheduled report for the specified account, and optionally, for all sub-accounts.
<u>acct.search</u>	Gets a list containing account information for the specified parent account and the client accounts that the API user has permission to access.
<u>acct.status.set</u>	Sets the status of the specified account.
<u>subacct.new</u>	Adds a new client account for the specified parent account.

[acct.category.get](#)

Gets the category of the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.category.get",
  "params": [
    "account_id"
  ]
}
```

Request parameters

account_id

Required string. The unique, system-generated account ID. You can use ["acct.list" on page 59](#) or ["acct.search" on page 78](#) to get the account ID.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "string"
}
```

Response parameters

result

String. The category of the specified account.

Permissions

The API user must be a member of the specified account.

Remarks

An account category is a user-defined label that you create via the Call Analytics UI on a parent account. You can use categories to logically group subaccounts for business purposes. For example, on the parent account, there are two account categories defined, "Local" and "Other states." You can call ["acct.category.set" on the facing page](#) to categorize your subaccounts according to whether they apply locally, or to other states in the country.

Example

The following example shows a successful call to the **acct.category.get**. The response returns the account category, "Automotive", for the account specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.category.get",
  "params": [
    "QrOXxEE9-fATtgAD"
  ]
}
```

```
}
]
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "Automotive"
}
```

acct.category.set

Sets the category of the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.category.set",
  "params": [
    "account_id",
    "category"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

account_id

Required string. The unique, system-generated account ID. You can use ["acct.list" on page 59](#) or ["acct.search" on page 78](#) to get the account ID.

category

Required string, which can contain up to 64 characters. The category for the specified account. The string should match an existing category configured on the parent account.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must have Account Admin permissions in the specified account.



Note: API users created in the parent account must have the Client Admin permission.

Remarks

An account category is a user-defined label that you create via the Call Analytics UI on a parent account. You can use categories to logically group sub-accounts for business purposes. For example, on the parent account, there are two account categories defined, "Local" and "Other states." You can call **acct.category.set** to categorize your sub-accounts according to whether they apply locally, or to other states in the country.

Example

The following example shows a successful call to the **acct.category.set**. The request sets the account category to "Automotive" for the specified account.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.category.set",
  "params": [
    "QrOXxEE9-fATtgAD",
    "Automotive"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

See also

["acct.category.get" on page 38](#)

acct.custom.define

Defines an account custom field. You use this method to define the name and data type of the custom field, and you use [acct.custom.set](#) to specify the value for the custom field.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.custom.define",
  "params": [
    "account_id",
    field_number,
    "field_name",
    "field_type",
    [
      "menu_value_1",
      "menu_value_2",
      ...,
      "menu_value_n"
    ]
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

account_id

Required string. The unique, system-generated account ID of the account that you want to configure. You can use ["acct.list" on page 59](#) or ["acct.search" on page 78](#) to get the account ID.

field_number

Required integer, which can contain a value between 1 and 20. The index number of the custom field.

field_name

Required string, which can contain up to 32 characters. The name of the custom field. The name is used to identify the custom field. The name is displayed in the user interface, and you specify the name when programmatically setting the value of the field with the **acct.custom.set** method.

Note The custom field name must be unique across the entire account.

field_type

Required string. The type of the custom field. The custom field type restricts what values are allowed to be set for the custom field in the user interface and via **acct.custom.set**. The following table describes the supported data types:

Value	Description
bool	A Boolean value. A drop-down list containing "yes" and "no" is displayed in the user interface, from which a user can select a value.
date	A date value. A date picker is displayed in the user interface, from which a user can select a date value.
float	A floating-point decimal numeric value. A text box is displayed in the user interface, in which a user can provide a floating-point decimal value.

Value	Description
int	An integer numeric value. A text box is displayed in the user interface, in which a user can provide an integer numeric value.
menu	An array of strings that you specify in the <i>menu items array</i> . A drop-down list containing the strings specified in the array is displayed in the user interface, from which a user can select a value.
text	A free-form text value. A text box is displayed in the user interface, in which a user can provide a free-form text value.

menu items array

Required array. If *field_type* is anything other than **menu**, specify an empty array. For the **menu field_type**, specify an array of strings. Each string in the array is displayed in the user interface as an item in the menu's drop-down list. If you set a value for the menu via **acct.custom.set**, the string you specify in **acct.custom.set** must exactly match (including case) one of the strings that you have specified in this array.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

You use account custom fields to support business-specific information. You can define up to 20 custom fields per account.

Inheriting custom fields

By default, a client account inherits the account custom fields from its parent account, and you see the inherited fields in the user interface on the Company Info and Settings page of the client account. If you use **acct.custom.define** or the user interface to define an account custom field for a client account, the client account no longer inherits account custom fields from the parent account, and only the custom field defined on the client account will be displayed on the Company Info and Settings page of the client account.

Examples

The following JSON-RPC requests and responses describe successful calls to the **acct.custom.define** method.

Example 1: Boolean custom field

The following request defines the name of account custom field 1 to **Account Reviewed?**, and "bool" as the type. The name of the custom field is displayed in the user interface next to a drop-down list containing "yes" and "no."

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.custom.define",
  "params": [
    "QrOXxEE9-fATtgAD",
    1,
    "Account Reviewed?",
    "bool",
    []
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

Example 2: Menu custom field

The following request defines the name of account custom field 2 to **Account Region**, which is a menu. The name of the custom field is displayed in the user interface, and each string in the array is displayed as an item in the menu's drop-down list.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.custom.define",
  "params": [
    "QrOXxEE9-fATtgAD",
    2,
    "Account Region",
    "menu",
    [
      "Central",
      "East North Central",
      "Northeast",
      "Northwest",
      "South",
      "Southeast",
      "Southwest",
      "West",
      "West North Central"
    ]
  ]
}
```

```
]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

acct.custom.get

Retrieves the value of the account custom fields.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.custom.get",
  "params": [
    "account_id"
  ]
}
```

Request parameters

account_id

Required string. The unique, system-generated account ID.

Response syntax

```
{
  "jsonrpc" : "2.0"
  "id" : "1",
  "result": [

    {
      fields": {
        "1": {
          "Industry": "Industry1"
        },
        "2": {
          "Integer Test": "12345"
        },
        "3": {
          "acc_custom3": ""
        },
        "4": {
```

```

        "acc_custom4": ""
    },
    "5": {
        "acc_custom5": ""
    },
    "6": {
        "acc_custom6": ""
    },
    "7": {
        "acc_custom7": ""
    },
    "8": {
        "acc_custom8": ""
    },
    "9": {
        "acc_custom9": ""
    },
    "10": {
        "acc_custom10": ""
    },
    "11": {
        "acc_custom11": ""
    },
    "12": {
        "acc_custom12": ""
    },
    "13": {
        "acc_custom13": ""
    },
    "14": {
        "acc_custom14": ""
    },
    "15": {
        "acc_custom15": ""
    },
    "16": {
        "acc_custom16": ""
    },
    "17": {
        "acc_custom17": ""
    },
    "18": {
        "acc_custom18": ""
    },
    "19": {
        "acc_custom19": ""
    },
    "20": {
        "acc_custom20": ""
    }
}

```

```
    }
  ],
}
```

Response parameters

result

The result contains a label representing the field name and a field value (if assigned) with the following format:

```
"field label" : "field value"
  field
```

Required string. The label of the custom field.

```
  value
```

Required string. The value of the account custom field. The value must be appropriate to the field type of the specified custom field.

Permissions

The API user must have Account Admin permissions in the specified account.

Example

Example Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.custom.get",
  "params": [
    "account_id"
  ]
}
```

Example Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "account_id": "account_id",
      "fields": {
        "1" {
          "acc_custom1": ""
        },
        "2" {
          "acc_custom2": ""
        },
        "3" {
          "acc_custom3": ""
        },
      }
    }
  ]
}
```

```
"4" {
  "acc_custom4": ""
},
"5" {
  "acc_custom5": ""
},
"5" {
  "acc_custom5": ""
},
"6" {
  "acc_custom6": ""
},
"7" {"acc_custom7": ""
},
"8" {
  "acc_custom8": ""
},
"9" {
  "acc_custom9": ""
},
"10" {
  "acc_custom10": ""
},
"11" {
  "acc_custom11": ""
},
"12" {
  "acc_custom12": ""
},
"13" {
  "acc_custom13": ""
},
"14" {
  "acc_custom14": ""
},
"15" {
  "acc_custom15": ""
},
"16" {
  "acc_custom16": ""
},
"17" {
  "acc_custom17": ""
},
"18" {
  "acc_custom18": ""
},
"19" {
  "acc_custom19": ""
}
```



```

    },
    "20" {
        "acc_custom20": ""
    }
}
],
}

```

acct.custom.set

Sets the value of the specified account custom field.

Request syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.custom.set",
  "params": [
    "account_id",
    "field_name",
    "field_value"
  ]
}

```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

account_id

Required string. The unique, system-generated account ID of the account that you want to configure. You can use ["acct.list" on page 59](#) or ["acct.search" on page 78](#) to get the account ID.

field_name

Required string. The name of the custom field to configure.

field_value

Required. The value of the campaign custom field. The value must be appropriate to the field type of the specified custom field, as described in the following table:

Field Type	Values
bool	A string with one of the following values: <ul style="list-style-type: none"> » "yes" or "1" » "no" or "0"

Field Type	Values
date	<p>A date string in the format "DD Mon YYYY" where:</p> <ul style="list-style-type: none"> » DD is a two-digit day. » Mon is the first three letters of the month. (Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec) » YYYY is the four-digit year <p>For example: "24 Mar 2016"</p>
float	<p>A floating-point decimal numeric value, for example: 3.14159</p> <p>Note you may also specify the value as a string, for example: "5.7"</p>
int	<p>An integer numeric value. For example: 155</p> <p>Negative integers are allowed. For example: -33</p> <p>Note you may also specify the value as a string, for example: "161"</p>
menu	<p>A string that matches exactly (including case) one of the strings defined for the menu.</p> <p>If you used "acct.custom.define" on page 41 to define the custom field, the value must be one of the strings that was specified in the <i>menu items array</i> request parameter.</p>
text	A string with a maximum length of 64 characters.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

You must use the name of the account custom field to set its value. You cannot use the ordinal position of the custom field.

Example

The following JSON-RPC request and response describe a successful call to the **acct.custom.set** method. The request sets the value of the account custom field named **Account Reviewed?** to "no" for the specified account.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.custom.set",
  "params": [
    "QrOXxEE9-fATtgAD",
    "Account Reviewed?",
    "no"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

acct.customid.get

Gets the custom reference ID for the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.customid.get",
  "params": [
    "account_id"
  ]
}
```

Request parameters

account_id

Required string. The unique, system-generated account ID.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "custom ref id"
}
```

Response parameters

result

A string containing the custom reference ID for the specified account.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

This method gets the custom reference ID of the specified account. The custom reference ID is also available in the user interface, on the **Company Info and Settings** page, under the **Account** tab.

A custom reference ID is a user-defined label for an account, which can be used to identify an account for business purposes.

Example

The following JSON-RPC request and response show a successful call to the **acct.customid.get** method. The response returns the custom reference ID, **CustomID-000001**, of the account specified in the request.

Example Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.customid.get",
  "params": [
    "QrOXxEE9-fATtgAD"
  ]
}
```

Example Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "CustomID-000001"
}
```

acct.customid.set

Sets the custom reference ID for the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.customid.set",
  "params": [
    "account_id",
    "custom_ref_id"
  ]
}
```

```
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

account_id

Required string. The unique, system-generated account ID of the account that you want to configure.

custom_ref_id

Required string. The custom reference ID for the specified account.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

This method sets the custom reference ID of the specified account. The custom reference ID is also available in the user interface, from the **Company Info and Settings** page, under the **Account** tab.

A custom reference ID is a user-defined label for an account, which can be used to identify an account for business purposes.

Example

The following JSON-RPC request and response show a successful call to the **acct.customid.set** method. The request sets the custom reference ID for the specified account to "CustomID-000001".

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.customid.set",
  "params": [
    "QrOXxEE9-fATtgAD",
    "CustomID-000001"
  ]
}
```

```
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

acct.keyword_spotting.word_list.get

Gets the list of call transcription keywords that have been set for each speaker (the agent and/or caller) on the specified account. You can set keywords for the agent or the caller using the Call Analytics user interface or the "[acct.keyword_spotting.word_list.set](#)" on [page 57](#) method.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.keyword_spotting.word_list.get",
  "params": [
    {
      "account_id": "string"
    }
  ]
}
```

Request parameters

The *params* array requires one (and only one) object with the following key-value pairs:

account_id

Required string. The unique, system-generated ID of the account that you want to get keywords for. You can use "[acct.list](#)" on [page 59](#) to get the account ID.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "keywords": {
      "agent": ["string", "string", ...],
      "caller": ["string", "string", ...]
    },

```

```

    "source": "string"
  }
}

```

Response parameters

The result object contains the following key-value pairs:

keywords

An object that contains 0, 1, or 2 key-value pairs depending if keywords are set, as follows:

- » If no keywords have not been set on the specified account, the *keywords* object is empty.
- » If keywords have been set for only for one of the speakers (agent and/or caller), the keywords object contains only the list of keywords that have been set for the speaker, in the *agent* array or *caller* array, as applicable.
- » If keywords have been set for both speakers, the *keywords* object contains both the *agent* and *caller* arrays of keywords.

Note the following:

- » The total number of keywords that can be returned is 300 per speaker.
- » Each keyword is a string that can be a single word or a phrase that contains at most 3 words separated by a space.

source

Either **account** or **default**. If the specified account ID in the request is a parent account, the *source* will always be **account**. If the specified account ID is a client account:

- » If the client account inherits keywords from its parent account, the *source* is **default**.
- » If the client account has keywords of its own set, the *source* is **account**.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

Call transcription must be enabled on the account for keywords to have been set. If call transcription is not enabled on the account, an error is returned. Contact Client Services to get call transcription enabled.

When keywords have been set on an account, Call Analytics searches call transcriptions and flags calls that contain any of the keywords. To get a list of calls that contain one or more keywords, use the ["Response Syntax: Basic Response" on page 253](#) method.

Example1

In this example, the keywords object in the response is empty, which means that no keywords have been set on the account.

Request

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.keyword_spotting.word_list.get",
  "params": [
    {

```

```
        "account_id": "QrOXxEE9-fATtgAD"
    }
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "keywords": {},
    "source": "account"
  }
}
```

Example 2

In this example, the keywords object in the response contains only a list of keywords for the agent. The source is set to default, which means the specified account in the request is a client account that inherited the keywords from its parent account.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.keyword_spotting.word_list.get",
  "params": [
    {
      "account_id": "fwABAVWwcP9gfwDk"
    }
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "keywords": {
      "agent": ["credit card", "expiration date",
        "security code"]
    },
    "source": "default"
  }
}
```



```
}
```

acct.keyword_spotting.word_list.set

Sets the specified list of call transcription keywords for each speaker (the agent and/or caller) on the specified account.

Note: The **acct.keyword_spotting.word_list.set** method replaces all the keywords currently set on the account with the keywords that you specify. If you want to add keyword(s) to those that are already set:

1. Call **acct.keyword_spotting.word_list.get**.
2. Add the keyword(s) to the applicable list(s) in the returned keywords object.
3. Specify the updated keywords object in the request to **acct.keyword_spotting.word_list.set**.

Request syntax

Required parameters are in bold.

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.keyword_spotting.word_list.set",
  "params": [
    {
      "account_id": "string",
      "keywords": {
        "agent": ["string1", "string2", ... ],
        "caller": ["string1", "string2", ... ]
      }
    }
  ]
}
```

Request parameters

The *params* array requires one (and only one) object with the following key-value pairs:

account_id

Required string. The unique, system-generated ID of the account that you want to set keywords for. You can use **"acct.list"** on page 59 to get the account ID.

keywords

Required object. The object contains the list of keywords for each speaker (*agent* and/or *caller*) that you want to set. If you want to set keywords for only one speaker, do not include the list for the other speaker. If you want to remove all keywords that have been set for an account, include an empty keywords object in the request.

Note the following:

- » Each keyword is a string that can be a single word or a phrase that contains at most 3 words separated by a space. If you include more than 3 words in a string, none of the keywords are set and an error is returned.
- » The total number of keywords that you can set is 300 per speaker.

- » Do not use special characters such as "%" or "&" or punctuation marks such as "?" or "!" in a keyword. Except for apostrophes (for example, **didn't** or **car's**), call transcriptions do not contain special characters or punctuation, so keywords containing these characters will not match anything in the transcriptions.
- » Do not use numbers in a keyword because the numbers may get redacted in the call transcriptions.
- » The search to find keywords in a call transcription is case insensitive, so the keyword **CREDIT CARD** matches **credit card** in the transcriptions.
- » The keyword or phrase must be an exact match. For example, the keyword **car** will not match **cars** or **car's** in the transcriptions.
- » Duplicate keywords are not allowed per speaker and are not set.
- » You may want to set the same keyword for both the agent and the caller to increase the odds that the keyword is found in the transcriptions. Of course, this depends on what the keyword is. For example, it makes sense to configure **credit card** as a keyword for both the agent and caller. But when you add a keyword for a promotional sale or additional service that you have trained your agents to mention to callers, you would configure the keyword only for the agent.

Response syntax

A successful response looks like the following:

```
{
  "jsonrpc":
    "2.0",
  "id":1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

- » After keywords have been set on an account, Call Analytics flags the keywords in transcriptions of new incoming calls. To get a list of calls that contain one or more keywords, use the ["Response Syntax: Basic Response" on page 253](#) method.
- » Keywords are not spotted retroactively. Only calls that arrive after the keywords have been set will have keywords flagged in the transcriptions.
- » Call transcription must be enabled on the account to set keywords. If call transcription is not enabled on the account, an error is returned. Contact Client Services to get call transcription enabled.

Example

The following request shows setting keywords for both the agent and the caller.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.keyword_spotting.word_list.set",
  "params": [
    {
      "account_id": "QrOXxEE9-fATtgAD",
      "keywords": [
        {
          "agent": ["credit card", "expiration date"],
          "caller": ["credit card"]
        }
      ]
    }
  ]
}
```

Response

```
{
  "jsonrpc":
    "2.0",
  "id": 1,
  "result": 1
}
```

acct.list

Gets a list containing account information using the account that the API user is a member of. Typically, the API user is a member of a parent account, and **acct.list** returns the parent account and all client accounts, including accounts that are disabled. If you want a list that is restricted to the parent account and only the client accounts that the API user has permission to access, use ["acct.search" on page 78](#), which also allows you to filter the list by account status, name, and Custom Ref. ID.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.list",
  "params": [
  ]
}
```

Request parameters

This method does not take any request parameters, but you still must provide the *params* array in the request even though it is empty. The method uses the account ID of the API user to get the list of accounts.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "acct": "string",
      "customid": "string",
      "name": "string",
      "status": "string"
    }
  ]
}
```

Response parameters

The result is an array that contains 1 or more objects, each of which contains the following key-value pairs:

acct

String. The unique, system-generated ID of the account.

customid

String. The value in the Custom Ref. ID field from the Company Info and Settings page in the Call Analytics user interface. An empty string ("") is returned if the account does not have a Custom Ref. ID.

name

String. The name of the account.

status

String. The status of the account: **active** or **disabled**.

Permissions

The API user must have Account Admin permissions.

Remarks

The method does not require any parameters, because the user credentials provided in the HTTP header are used to determine what account information is returned by the method.

Example

The following JSON-RPC request and response describe a successful call to the **acct.list** method. The parent account that the API user is a member of and all client accounts are returned in the response.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.list"
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
```

```

    "result": [
      {
        "acct": "CA6ph0qob-ZxRgJP",
        "customid": "",
        "status": "active",
        "name": "Test account 1"
      },
      {
        "acct": "CA6ph0qob-ZxRgFA",
        "customid": "",
        "status": "active",
        "name": "Test account 2"
      },
      {
        "acct": "CA6ph0qob-ZxTYJP",
        "customid": "",
        "status": "active",
        "name": "Test account 3"
      },
      {
        "acct": "CA6ph0qob-ZYaaJR",
        "customid": "",
        "status": "active",
        "name": "Test account 4"
      }
    ]
  }

```

acct.membership.list

Gets a list containing account membership information for the specified user.

Request syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.membership.list",
  "params": [
    "uid"
  ]
}

```

Request parameters

uid

Required string, which can contain exactly 16 characters. The unique, system-generated user ID of the specified user.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      account_admin,
      "account_id",
      "status",
      report_admin
    }
  ]
}
```

Response parameters

: Optional object.

account_admin

Optional boolean. **true** if the user is an account administrator for the account; otherwise, **false**.

account_id

Optional string, which can contain exactly 16 characters. The unique, system-generated account ID of the account.

report_admin

Optional boolean. **true** if the user is a report administrator for the account; otherwise, **false**.

result

Required array, which can contain at least 0 items. An array of objects, each of which represents account membership information for an account.

status

Optional string. The name of the account.

Permissions

The user must have administrator permissions.

Example

The following JSON-RPC request and response describe a successful call to the **acct.membership.list** method. The response returns basic information about the client accounts to which the user specified in the request is associated.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.membership.list",
  "params": [
    "CmymDlNNgSxFmWd-"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
```

```

    "id": 1,
    "result": [
      {
        "account_admin":true,
        "account_name":"Test client account 1",
        "account_id":"CmymM1NVgl0qwgDC",
        "report_admin":true
      },
      {
        "account_admin":true,
        "account_name":"Test client account 2",
        "account_id":"CA6phkqC8sky6wGD",
        "report_admin":true
      },
      {
        "account_admin":false,
        "account_name":"Test client account 3",
        "account_id":"CA6phEswDdiMxQFj",
        "report_admin":false
      }
    ]
  }
}

```

acct.membership.set

Sets the account membership information for the specified user and account.

Request syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.membership.set",
  "params": [
    "accid",
    "uid",
    account_admin,
    report_admin
  ]
}

```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

accid

Required string, which can contain exactly 16 characters. The unique, system-generated account ID of the specified account.

uid

Required string, which can contain exactly 16 characters. The unique, system-generated user ID of the specified user.

account_admin
Required boolean. **true** if the specified user is an account administrator for the specified account; otherwise, **false**.

report_admin
Required boolean. **true** if the specified user is a report administrator for the specified account; otherwise, **false**.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result
Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions.

Example

The following JSON-RPC request and response describe a successful call to the **acct.membership.set** method. The request associates the specified user with the specified account, and indicates that the specified user should have both account administrator and report administrator permissions for the specified account.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.membership.set",
  "params": [
    "QrOXxEE9-fATtgAD",
    "CmymDlNNgSxFmWD-",
    true,
    true
  ]
}
```

Response


```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

acct.membership.unset

Unsets account membership information for the specified account and user.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.membership.unset",
  "params": [
    "accid",
    "uid"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

accid

Required string, which can contain exactly 16 characters. The unique, system-generated account ID of the specified account.

uid

Required string, which can contain exactly 16 characters. The unique, system-generated user ID of the specified user.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the specified account.

Remarks

This method does not delete the specified user, nor does it delete the specified account. The method just removes the association between the specified user and the specified account.

Example

The following JSON-RPC request and response describe a successful call to the **acct.membership.unset** method. The request removes membership for the specified user from the specified client account.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.membership.unset",
  "params": [
    "QrOXxEE9-fATtgAD",
    "CmymDlNNgSxFmwD-"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

acct.name.get

Gets the name of the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.name.get",
  "params": [
    "account_id"
  ]
}
```

Request parameters

account_id

Required string. The unique, system-generated account ID. You can use ["acct.list"](#) on page 59 or ["acct.search"](#) on page 78 to get the account ID.

Response syntax

```
{
  "jsonrpc": "2.0",
```

```
"id": 1,  
"result": "string"  
}
```

Response parameters

result

A string that contains the name of the specified account.

Permissions

The API user must be a member of the specified account.

Remarks

The account name is displayed in the Call Analytics user interface under the Account tab on the Company Info and Settings page.

Example

Request

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "method": "acct.name.get",  
  "params": [  
    "QrOXxEE9-fATtgAD"  
  ]  
}
```

Response

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "result": "Test account"  
}
```

See also

- » ["acct.name.set" below](#)
- » ["Accounts" on page 37](#)

acct.name.set

Sets the name of the specified account.

Request syntax

```
{  
  "jsonrpc": "2.0",  
  "id": 1,
```

```
    "method": "acct.name.set",
    "params": [
        "account_id",
        "name"
    ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

account_id

Required string. The unique, system-generated account ID. You can use ["acct.list" on page 59](#) to get the account ID.

name

Required string. The name for the specified account.

Response syntax

```
{
    "jsonrpc": "2.0",
    "id": 1,
    "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

The account name is displayed in the Call Analytics user interface under the Account tab on the Company Info and Settings page.

Example

The following example shows a successful call to the **acct.name.set** method.

Request

```
{
    "jsonrpc": "2.0",
    "id": 1,
    "method": "acct.name.set",
    "params": [
        "QrOXxEE9-fATtgAD",
        "Test account"
    ]
}
```

```
}
]
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

See also

- » ["acct.name.get" on page 66](#)
- » ["Accounts" on page 37](#)

acct.numlookup.set

Sets whether reverse lookup for incoming calls is enabled for the specified account

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.numlookup.set",
  "params": [
    "accid",
    numlookup
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

accid

Required string, which can contain exactly 16 characters. The unique, system-generated account ID of the specified account.

numlookup

Required boolean. **true** to enable reverse lookup for incoming calls; otherwise, **false**.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the specified account.

Remarks

If the reverse lookup feature has been enabled, using this method enables or disables reverse lookup information retrieval for the specified account. Reports and other API calls, such as **call.search**, can return reverse lookup information if the reverse lookup feature has been enabled.

Example

The following JSON-RPC request and response describe a successful call to the **acct.numlookup.set** method. The request enables reverse lookup information retrieval for the specified account.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.numlookup.set",
  "params": [
    "QrOXxEE9-fATtgAD",
    true
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

acct.phone.get

Gets the telephone number for the contact information of the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.phone.get",
  "params": [
    "accid"
  ]
}
```

```
}
}
```

Request parameters

accid

Required string, which can contain exactly 16 characters. The unique, system-generated account ID of the specified account.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "result"
}
```

Response parameters

result

Required string. The telephone number for the contact information of the specified account.

Permissions

The user must have administrator permissions for the specified account.

Remarks

This method gets the telephone number of the company address for the specified account. The company address is also available in the user interface, from the **Company Info and Settings** page, under the **Account** tab.

Example

The following JSON-RPC request and response describe a successful call to the **acct.phone.get** method. The response returns the contact telephone number, 800-555-1212, for the account specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.phone.get",
  "params": [
    "QrOXxEE9-fATtgAD"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "8005551212"
}
```

acct.phone.set

Sets the telephone number for the contact information of the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.phone.set",
  "params": [
    "accid",
    "phone"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

accid

Required string, which can contain exactly 16 characters. The unique, system-generated account ID of the specified account.

phone

Required string, which can contain up to 10 characters. The telephone number for the contact information of the specified account.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the specified account.

Remarks

This method sets the telephone number of the company address for the specified account. The company address is also available in the user interface, from the **Company Info and Settings** page, under the **Account** tab.

Example

The following JSON-RPC request and response describe a successful call to the **acct.phone.set** method. The request sets the telephone number of the company address for the specified account to 800-555-1212.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.phone.set",
  "params": [
    "QrOXxEE9-fATtgAD",
    "8005551212"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

acct.report.scheduled.delete

Deletes a recipient from a scheduled report. When all recipients are deleted from a scheduled report, in effect, the report is deleted.

When a scheduled report has only one user who receives an email of the report, you only need to call **acct.report.scheduled.delete** once to delete the scheduled report. When a scheduled report has multiple users who receives an email of the report, one call to **acct.report.scheduled.delete** removes one recipient from the scheduled report, but the other recipients still get the report. See the Remarks section below for additional information.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.report.scheduled.delete",
  "params": [
    {
      "sched_report_id": "string"
    }
  ]
}
```

Request parameters

The *params* array requires one (and only one) object with the following key-value pair:

sched_report_id

Required string. The system-generated ID that uniquely identifies the scheduled report and the recipient of the report. You obtain the *sched_report_id* by calling ["acct.report.scheduled.list"](#) on page 75.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must have Account Admin permissions on the account that the specified report was created in.

Remarks

You call **acct.report.scheduled.list** to get the *sched_report_id* value to pass to **acct.report.scheduled.delete**. When a report is scheduled to be sent to more than one user, the result array returned by **acct.report.scheduled.list** contains an object for each user who receives the report. In this case, each object in the list has a *sched_report_id* value that uniquely identifies the report and recipient of the report.

When a scheduled report has only one user who receives an email of the report, **acct.report.scheduled.delete** deletes the specified scheduled report. When a scheduled report has multiple users who receives an email of the report, one call to **acct.report.scheduled.delete** removes a recipient from the scheduled report; the other recipients still receive the report. To delete a report with multiple recipients, you must remove each recipient.

There are three use cases for **acct.report.scheduled.delete**:

1. To delete a scheduled report.
 - a. In the results array from **acct.report.scheduled.list**, look for the *report_title* that you want to delete.
 - b. Get all of the *sched_report_id* values for the target report.
 - c. For each *sched_report_id* value, call **acct.report.scheduled.delete**.
2. To remove a recipient from all scheduled reports. For example, you may want to do this when a report (or reports) is sent to a user who has left the company.
 - a. In the results array from **acct.report.scheduled.list**, look for the *user_email* of the recipient that you want to remove.
 - b. Get all of the *sched_report_id* values for the target user.
 - c. For each *sched_report_id* value, call **acct.report.scheduled.delete**.
3. To remove a recipient from a specific scheduled report. For example, a user may want to be removed from a report for one specific client, but still get reports on other clients.
 - a. In the results array from **acct.report.scheduled.list**, look for the target *report_title* and *user_email*, and get the corresponding *sched_report_id* value.
 - b. Using that *sched_report_id* value, call **acct.report.scheduled.delete**.

Example

The following JSON-RPC request and response show a successful call to the **acct.report.scheduled.delete** method.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.report.scheduled.delete",
  "params": [
    {
      "sched_report_id": "CmyMRZSIyRUqGACD"
    }
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

acct.report.scheduled.list

Returns information about all scheduled reports for the specified account, and optionally, information about scheduled reports for sub-accounts.

Request syntax

Required parameter is in bold.

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.report.scheduled.list",
  "params": [
    {
      "account_id": "string",
      "subacct": boolean
    }
  ]
}
```

Request parameters

The *params* array requires one (and only one) object with the following key-value pairs:

account_id

Required string. The unique, system-generated account ID of the account you want to get scheduled reports for.

subacct

Optional boolean. If not included or **false**, information about the scheduled reports for only the specified account is returned. If **true**, information about the scheduled reports for the specified account and for all of its sub-accounts is returned.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "account_id": "account ID of user receiving report",
      "frequency": "scheduled frequency",
      "report_title": "report title",
      "report_type": "type of report",
      "sched_report_id": "ID for report and recipient",
      "target_account": "account ID that the report is for",
      "user_email": "recipient of report"
    }
  ]
}
```

Response parameters

The result is an array that contains 0 or more objects. Each object contains the following key-value pairs:

account_id

The system-generated account ID of the account that the user receiving the report was created in. If the user was created in a parent account, and is receiving an email report on a client account, the *account_id* and *target_account* will be different.

frequency

The frequency (Daily, Weekly, Monthly, Once) that the report is scheduled to run.

report_title

The title of the report.

report_type

The report type. The returned string describes the page in the user interface where the report was scheduled, for example: **/clients/report/dashboard**

sched_report_id

The system-generated ID that uniquely identifies the scheduled report and the recipient of the report. When a report is scheduled to be sent to more than one user, the result array contains an object for each user who receives the report. The *sched_report_id* value is unique in each object.

target_account

The system-generated account ID of the account that the scheduled report is reporting on.

user_email

The email address of the user who receives the report.

Permissions

- » The API user must have Account Admin permissions on the specified account.
- » When getting a list of reports for a parent account and all of its sub-accounts, the API user must have Account Admin and Client Admin permissions in the parent account. If the API user does not have the Client Admin permission, only the scheduled reports for the parent account are returned.

Remarks

When a report is scheduled to be sent to more than one user, the result array contains an object for each user who receives the report. In this case, each object in the list has a *sched_report_id* value that uniquely identifies the report and recipient of the report. This allows you to call **"acct.report.scheduled.delete"** on [page 73](#) to remove a specific recipient from a scheduled report.

Example

The following JSON-RPC request and response show a successful call to the **acct.report.scheduled.list** method to retrieve scheduled reports for the parent account and all sub-accounts. Three objects are returned in the results array. The first two objects are for the same report because the report is emailed to two different users. Note that even though the first two objects contain information on the same report, the *sched_report_id* is different.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.report.scheduled.list",
  "params": [
    {
      "account_id": "CmyMQVSPja9GqgBI",
      "subacct": true
    }
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "frequency": "Weekly",
      "sched_report_id": "CmyMQVTQTvgv7ACH",
      "target_account": "CmyMQVTQTI0uygDY",
      "report_type": "/campaigns/report/summary",
      "user_email": "asmith@examplecompany.com",
      "report_title": "Calls-Summary Report",
      "account_id": "CmyMQVSPjQtGhQCQ"
    },
    {
      "frequency": "Weekly",
      "sched_report_id": "CmyMQVTQTvgv7ACI",
      "target_account": "CmyMQVTQTI0uygDY",
      "report_type": "/campaigns/report/summary",
      "user_email": "bjones@examplecompany.com",
      "report_title": "Calls-Summary Report",
      "account_id": "CmyMQVSPjQtGhQCQ"
    },
    {
      "frequency": "Monthly",
      "sched_report_id": "CmyMQVTQUjgw1QAC",
      "target_account": "CmyMQVSPjQtGhQCQ",
      "report_type": "/campaigns/report/calls_by_time",
      "user_email": "jharris@examplecompany.com",
      "report_title": "Calls-by-Hour-and-Day-of-Week Report",
    }
  ]
}
```

```
        "account_id": "CmyMQVSPjQtGhQCQ"
    }
]
}
```

acct.search

Gets a list containing account information for the specified parent account and the client accounts that the API user has permission to access. You can filter this list by supplying one or more of the optional search parameters described below. If you include search parameter(s) in the request, all the criteria that you specify must match for the account to be included in the returned list. In other words, the method does a logical AND on the search criteria specified in the parameters.

Request syntax

Required parameters are in bold.

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.search",
  "params": [
    {
      "account_id": "string",
      "name": "string",
      "customid": "string",
      "status": "string"
    }
  ]
}
```

Request parameters

The *params* array requires one (and only one) object with the following key-value pairs:

account_id

Required string. The unique, system-generated ID of an account. Typically, you set this to the account ID of a parent account, but the method also accepts an account ID of a client account. You can use ["acct.list" on page 59](#) to get the account ID.

name

Optional string. If specified, the method does a case-insensitive search for account names that match the string you provide.

customid

Optional string. This is the Custom Ref. ID field from the Company Info and Settings page in the Call Analytics user interface. If specified, the method does a case-insensitive search for accounts that match the string you provide.

status

Optional string. To filter the search results based on account status, enter one of the following strings: **active**, **disabled**, or **all**. If the status is not specified, only active client accounts are returned.

You can use the following wildcard characters in the *name* and *customid* search strings both before and after the other characters in the string:

- » The percent sign ("%") is a substitute for 0 or more characters.
- » The underscore ("_") is a substitute for exactly 1 character.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "account_id": "string",
      "parent_account_id": "string",
      "name": "string",
      "customid": "string",
      "status": "string"
    }
  ]
}
```

Response parameters

The result is an array that contains 0 or more objects, each of which contains the following key-value pairs:

account_id

The unique, system-generated ID of the account.

name

The name of the account.

parent_account_id

The unique, system-generated ID of the parent account. If the returned account is a parent account, *parent_account_id* is an empty string.

customid

The value in the Custom Ref. ID field from the Company Info and Settings page in the Call Analytics user interface. An empty string ("") is returned if the account does not have a Custom Ref. ID.

status

The status of the account: **active** or **disabled**.

Permissions

- » The API user must be a member of the specified account.
- » If the API user has Client Viewer or Client Admin permissions in the specified parent account, the parent account and all client accounts are searched.
- » If the API user has neither Client Viewer nor Client Admin permissions in the specified parent account, the search is restricted to the parent account and only the client accounts that the API user has permission to access.

Remarks

As a best practice, if you do not want to specify search criteria, omit the search parameter (or parameters) in the request. Results may vary if you include a search parameter and set it to an empty string, null, or undefined.

Example

The following example request searches for accounts that match the string in the name parameter. Because the status parameter is not specified, only active accounts are returned.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.search",
  "params": [
    {
      "account_id": "QrOX7UWtgfg8mwYk",
      "name": "example%"
    }
  ]
}
```

Response

The returned list contains one client account and the parent account.

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "status": "active",
      "customid": "",
      "name": "Example Company - Client 1",
      "account_id": "CtjSZVRhQ0gdwACQ",
      "parent_account_id": "QrOX7UWtgfg8mwYk"
    },
    {
      "status": "active",
      "customid": "",
      "name": "Example Company",
      "account_id": "QrOX7UWtgfg8mwYk",
      "parent_account_id": ""
    }
  ]
}
```

acct.status.set

Sets the status of the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
```



```

    "method": "acct.status.set",
    "params": [
        "account_id",
        "status"
    ]
}

```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

account_id

Required string, which can contain exactly 16 characters. The unique, system-generated account ID of the specified account.

status

Required string. The status of the specified account. This property must be set to one of the following values:

Value	Description
active	The specified account is active.
disabled	The specified account is disabled.

Response syntax

```

{
    "jsonrpc": "2.0",
    "id": 1,
    "result": integer
}

```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

- » To change the status of a sub-account, the API user must have Account Admin and Client Admin permissions in the parent account that the sub-account belongs to.
- » To change the status of a parent account, contact Client Services.

Example

The following JSON-RPC request and response show a successful call to the **acct.status.set** method.

Request

```

{
    "jsonrpc": "2.0",
    "id": 1,
    "method": "acct.status.set",

```

```
    "params": [
      "QrOXxEE9-fATtgAD",
      "active"
    ]
  }
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

subacct.new

Creates a new sub-account for the specified parent account. Although the key-value pairs specifying the address and phone number are optional, an object for the contact information is required even if it is empty (see [Example 1: No Company Contact Info](#) below).

Request syntax

Required parameters are in bold. The object is required even if it is empty.

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "subacct.new",
  "params": [
    "account_id",
    "sub-account_name",
    {
      "city": "string",
      "country": "string",
      "phone": "string",
      "state": "string",
      "street1": "string",
      "street2": "string",
      "zip": "string"
    }
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

accountID

Required string. The unique, system-generated account ID of the parent account. You can use ["acct.list"](#) on [page 59](#) to get the account ID.

sub-account_name

Required string. The name for the new client account.

The required object contains the following key-value pairs:

city

Optional string. The city name.

country

Optional string. The 2-character ISO country code.

phone

Optional string. For the US and other countries in the North American Numbering Plan (NANP), this is a 10-digit phone number. When you supply a country code for a country not in the NANP (such as for Ireland), enter the phone number in E.164 format.

state

Optional string. The 2-character ISO state or region code.

street1

Optional string. The first line of the street address, which can contain up to 128 characters.

street2

Optional string. The second line of the street address, which can contain up to 128 character.

zip

Optional string. The postal or ZIP code, which can contain up to 10 characters.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "name": "client name",
    "acc": "client account id"
  }
}
```

Response parameters

The *result* object contains the following key-value pairs:

name

The name of the new client account.

acc

The unique, system-generated account ID of the new client account.

Permissions

The API user must have Account Admin and Client Admin permissions on the specified parent account.

Remarks

- » The new client account does not contain a default ad group. Call ["group.new"](#) on [page 115](#) to create a default ad group in the new client account.

- » The new client account inherits the value of the Allow Shutdown setting from the parent account. By default, Allow Shutdown is set to **Yes** in the parent account. Unless this setting was changed in the parent account, new client accounts will be created with Allow Shutdown set to **Yes**.

Example 1: No Company Contact Info

The following JSON-RPC request and response describe a successful call to the **subacct.new** method. Note the empty curly braces must be supplied in the request even though you are not specifying the company address or phone number..

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "subacct.new",
  "params": [
    "QrOXxEE9-fATtgAD",
    "Client Account One",
    {}
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "name": "Client Account One",
    "acc": "CmyGGlSwaHw3OwD9"
  }
}
```

Example 2: Company City and State

The following JSON-RPC request and response describe a successful call to the **subacct.new** method where the company's city and state are specified.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "subacct.new",
  "params": [
    "QrOXxEE9-fATtgAD",
    "Client Account Two",
    {
      "city": "Seattle",
      "state": "WA"
    }
  ]
}
```

```
}
]
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "name": "Client Account Two",
    "acc": "CmyGG1SwHzlCrADw"
  }
}
```

Account Addresses

The methods in this section allow you to get and set account addresses.

Method	Description
<u>acct.addr.city.get</u>	Gets the city name for the contact information of the specified account.
<u>acct.addr.city.set</u>	Sets the city name for the contact information of the specified account.
<u>acct.addr.country.get</u>	Gets the country or region code for the contact information of the specified account.
<u>acct.addr.country.set</u>	Sets the country or region code for the contact information of the specified account.
<u>acct.addr.postalcode.get</u>	Gets the postal code for the contact information of the specified account.
<u>acct.addr.postalcode.set</u>	Sets the postal code for the contact information of the specified account.
<u>acct.addr.state.get</u>	Gets the state or region code for the contact information of the specified account.
<u>acct.addr.state.set</u>	Sets the state or region code for the contact information of the specified account.
<u>acct.addr.street1.get</u>	Gets the first line of the street address for the contact information of the specified account.
<u>acct.addr.street1.set</u>	Sets the first line of the street address for the contact information of the specified account.
<u>acct.addr.street2.get</u>	Gets the second line of the street address for the contact information of the specified account.
<u>acct.addr.street2.set</u>	Sets the second line of the street address for the contact information of the specified account.

acct.addr.city.get

Gets the city name for the contact information of the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.city.get",
  "params": [
    "account_id"
  ]
}
```

Request parameters

account_id

Required string, which can contain exactly 16 characters. The unique, system-generated account ID of the specified account. You can use ["acct.list" on page 59](#) to get the account ID.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "result"
}
```

Response parameters

result

Required string. The city name for the contact information of the specified account.

Permissions

The user must have administrator permissions for the specified account.

Remarks

This method gets the city name of the company address for the specified account. The company address is also available in the user interface, from the **Company Info and Settings** page, under the **Account** tab.

Example

The following JSON-RPC request and response describe a successful call to the **acct.addr.city.get** method. The response returns the city name, "Seattle", of the company address for the account specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.city.get",
  "params": [
    "QrOXxEE9-fATtgAD"
  ]
}
```

```
]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "Seattle"
}
```

acct.addr.city.set

Sets the city name for the contact information of the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.city.set",
  "params": [
    "accid",
    "city"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

accid

Required string, which can contain exactly 16 characters. The unique, system-generated account ID of the specified account.

city

Required string, which can contain up to 64 characters. The city name for the contact information of the specified account.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the specified account.

Remarks

This method sets the city name of the company address for the specified account. The company address is also available in the user interface, from the **Company Info and Settings** page, under the **Account** tab.

Example

The following JSON-RPC request and response describe a successful call to the **acct.addr.city.set** method. The request sets the city name of the company address for the specified account to "Seattle".

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.city.set",
  "params": [
    "QrOXxEE9-fATtgAD",
    "Seattle"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

acct.addr.country.get

Gets the country or region code for the contact information of the specified account.

Request syntax


```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.country.get",
  "params": [
    "accid"
  ]
}
```

Request parameters

accid

Required string, which can contain exactly 16 characters. The unique, system-generated account ID of the specified account.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "result"
}
```

Response parameters

result

Required string. The country or region code for the contact information of the specified account.

Permissions

The user must have administrator permissions for the specified account.

Remarks

This method gets the country code of the company address for the specified account. The company address is also available in the user interface, from the **Company Info and Settings** page, under the **Account** tab.

Example

The following JSON-RPC request and response describe a successful call to the **acct.addr.country.get** method. The response returns the country code, "US", of the company address for the account specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.country.get",
  "params": [
    "QrOXxEE9-fATtgAD"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "US"
}
```

acct.addr.country.set

Sets the country or region code for the contact information of the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.country.set",
  "params": [
    "accid",
    "country"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

accid

Required string, which can contain exactly 16 characters. The unique, system-generated account ID of the specified account.

country

Required string, which can contain up to 2 characters. The country code for the contact information of the specified account.

This property should be set to the appropriate two-character ISO country code for the address.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the specified account.

Remarks

This method sets the country code of the company address for the specified account. The company address is also available in the user interface, from the **Company Info and Settings** page, under the **Account** tab.

Example

The following JSON-RPC request and response describe a successful call to the **acct.addr.country.set** method. The request sets the country code of the company address for the specified account to "US".

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.country.set",
  "params": [
    "QrOXxEE9-fATtgAD",
    "US"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

acct.addr.postalcode.get

Gets the postal code for the contact information of the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.postalcode.get",
  "params": [
    "accid"
  ]
}
```

Request parameters

accid

Required string, which can contain exactly 16 characters. The unique, system-generated account ID of the specified account.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "result"
}
```

Response parameters

result

Required string. The postal code for the contact information of the specified account.

Permissions

The user must have administrator permissions for the specified account.

Remarks

This method gets the postal code of the company address for the specified account. The company address is also available in the user interface, from the **Company Info and Settings** page, under the **Account** tab.

Example

The following JSON-RPC request and response describe a successful call to the **acct.addr.postalcode.get** method. The response returns the postal code, "98101", of the company address for the account specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.postalcode.get",
  "params": [
    "QrOXxEE9-fATtgAD"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "98101"
}
```

acct.addr.postalcode.set

Sets the postal code for the contact information of the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
```

```
"method": "acct.addr.postalcode.set",
"params": [
  "accid",
  "postalcode"
]
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

accid

Required string, which can contain exactly 16 characters. The unique, system-generated account ID of the specified account.

postalcode

Required string, which can contain up to 10 characters. The postal code for the contact information of the specified account.

This property should be set to the appropriate post code, postal code, or ZIP code, up to 10 characters, for the address.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the specified account.

Remarks

This method sets the postal code of the company address for the specified account. The company address is also available in the user interface, from the **Company Info and Settings** page, under the **Account** tab.

Example

The following JSON-RPC request and response describe a successful call to the **acct.addr.postalcode.set** method. The request sets the postal code of the company address for the specified account to "98101".

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
```

```
{
  "method": "acct.addr.postalcode.set",
  "params": [
    "QrOXxEE9-fATtgAD",
    "98101"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

acct.addr.state.get

Gets the state or region code for the contact information of the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.state.get",
  "params": [
    "accid"
  ]
}
```

Request parameters

accid

Required string, which can contain exactly 16 characters. The unique, system-generated account ID of the specified account.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "result"
}
```

Response parameters

result

Required string. The state or region code for the contact information of the specified account.

Permissions

The user must have administrator permissions for the specified account.

Remarks

This method gets the state code of the company address for the specified account. The company address is also available in the user interface, from the **Company Info and Settings** page, under the **Account** tab.

Example

The following JSON-RPC request and response describe a successful call to the **acct.addr.state.get** method. The response returns the state code, "WA", of the company address for the account specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.state.get",
  "params": [
    "QrOXxEE9-fATtgAD"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "WA"
}
```

acct.addr.state.set

Sets the state or region code for the contact information of the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.state.set",
  "params": [
    "accid",
    "state"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

accid

Required string, which can contain exactly 16 characters. The unique, system-generated account ID of the specified account.

state

Required string, which can contain up to 2 characters. The state or region code for the contact information of the specified account.

This property should be set to the appropriate 2-character ISO state or region code for the address.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the specified account.

Remarks

This method sets the state code of the company address for the specified account. The company address is also available in the user interface, from the **Company Info and Settings** page, under the **Account** tab.

Example

The following JSON-RPC request and response describe a successful call to the **acct.addr.state.set** method. The request sets the state code of the company address for the specified account to "WA".

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.state.set",
  "params": [
    "QrOXxEE9-fATtgAD",
    "WA"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```


acct.addr.street1.get

Gets the first line of the street address for the contact information of the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.street1.get",
  "params": [
    "accid"
  ]
}
```

Request parameters

accid

Required string, which can contain exactly 16 characters. The unique, system-generated account ID of the specified account.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "result"
}
```

Response parameters

result

Required string. The first line of the street address for the contact information of the specified account.

Permissions

The user must have administrator permissions for the specified account.

Remarks

This method gets the first street address line of the company address for the specified account. The company address is also available in the user interface, from the **Company Info and Settings** page, under the **Account** tab.

Example

The following JSON-RPC request and response describe a successful call to the **acct.addr.street1.get** method. The response returns the first street address line, "123 Street Road", from the company address for the account specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.street1.get",
  "params": [
```

```
    "QrOXxEE9-fATtgAD"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "123 Street Road"
}
```

acct.addr.street1.set

Sets the first line of the street address for the contact information of the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.street1.set",
  "params": [
    "accid",
    "street1"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

accid

Required string, which can contain exactly 16 characters. The unique, system-generated account ID of the specified account.

street1

Required string, which can contain up to 128 characters. The first line of the street address for the contact information of the specified account.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the specified account.

Remarks

This method sets the first street address line of the company address for the specified account. The company address is also available in the user interface, from the **Company Info and Settings** page, under the **Account** tab.

Example

The following JSON-RPC request and response describe a successful call to the **acct.addr.street1.set** method.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.street1.set",
  "params": [
    "QrOXxEE9-fATtgAD",
    "123 Street Road"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

acct.addr.street2.get

Gets the second line of the street address for the contact information of the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.street2.get",
  "params": [
    "accid"
  ]
}
```

Request parameters

accid

Required string, which can contain exactly 16 characters. The unique, system-generated account ID of the specified account.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "result"
}
```

Response parameters

result

Required string. The second line of the street address for the contact information of the specified account.

Permissions

The user must have administrator permissions for the specified account.

Remarks

This method gets the second street address line of the company address for the specified account. The company address is also available in the user interface, from the **Company Info and Settings** page, under the **Account** tab.

Example

The following JSON-RPC request and response describe a successful call to the **acct.addr.street2.get** method. The response returns the second street address line, "Suite A113", of the company address for the account specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.street2.get",
  "params": [
    "QrOXxEE9-fATtgAD"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "Suite A113"
}
```

acct.addr.street2.set

Sets the second line of the street address for the contact information of the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.street2.set",
  "params": [
    "accid",
    "street2"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

accid

Required string, which can contain exactly 16 characters. The unique, system-generated account ID of the specified account.

street2

Required string, which can contain up to 128 characters. The second line of the street address for the contact information of the specified account.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the specified account.

Remarks

This method sets the second street address line of the company address for the specified account. The company address is also available in the user interface, from the **Company Info and Settings** page, under the **Account** tab.

Example

The following JSON-RPC request and response describe a successful call to the **acct.addr.street2.set** method. The request sets the second street address line of the specified account to "Suite A113".

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "acct.addr.street2.set",
  "params": [
    "QrOXxEE9-fATtgAD",
    "Suite A113"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

User Settings

This section describes the methods available to manage users.

Method	Description
"user.del" below	Deletes the specified user.
"user.email.set" on page 104	Sets the email address of the specified user.
"user.list" on page 105	Lists the active users associated with the specified account.
"user.new" on page 107	Adds a new user to the specified account.
"user.passwd.set" on page 108	Sets the password of the specified user.
"user.permissions.get" on page 110	Get the permissions for a user for an account.
"user.permissions.set" on page 111	Set the permissions for a user for an account.

user.del

Deletes the specified user.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "user.del",
}
```

```
    "params": [
      "user_id"
    ]
  }
```

Request parameters

user_id

Required string. The unique, system-generated ID of the user you want to delete. You can use ["user.list"](#) on [page 105](#) to get the *user_id*.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must have Account Admin permission in the account in which the specified user was created in.

Remarks

The response is not instantaneous because before the user is deleted, the user must be removed from all ad groups, and parent account users must be removed from all sub-accounts that it is a member of.

Example

The following JSON-RPC request and response show a successful call to the **user.del** method.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "user.del",
  "params": [
    "Cmymc1NNy2Av0gB1"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
```

```
"result": 1
}
```

user.email.set

Sets the email address of the specified user.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "user.email.set",
  "params": [
    "uid",
    "eml"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

eml

Required string. The email address of the specified user.

uid

Required string. The unique, system-generated user ID of the specified user.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified user is associated.

Example

The following JSON-RPC request and response describe a successful call to the **user.email.set** method. The method sets the email address of the specified user to "someone@somewhere.com".

Request


```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "user.email.set",
  "params": [
    "CmymclNNy2Av0gB1",
    "someone@somewhere.com"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

user.list

Lists the active users in the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "user.list",
  "params": [
    "account_id"
  ]
}
```

Request parameters

account_id

Required string. The unique, system-generated ID of the account that the users belong to.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "email": "string",
      "name": "string",
      "uid": "string"
    },
    ...
  ]
}
```

```

    {
      "email": "string",
      "name": "string",
      "uid": "string"
    }
  ]
}

```

Response parameters

The result is an array that contains 0 or more objects. Each object contains the following key-value pairs:

email

A string containing the email address of the user.

name

A string containing the name of the user.

uid

A string containing the unique, system-generated user ID of the user.

Permissions

The API user must have Account Admin permission in the account in which the specified user was created in.

Example

The following JSON-RPC request and response show a successful call to the **user.list** method. The response returns information about the active users in the account specified in the request.

Request

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "user.list",
  "params": [
    "QrOXxEE9-fATtgAD"
  ]
}

```

Response

```

{
  "jsonrpc": "2.0",
  "id": 32860,
  "result": [
    {
      "email": "jsmith@examplecompany.com",
      "uid": "Cmymc1NNy2Av0gB1",
      "name": "John Smith"
    },
    {
      "email": "mbrown@examplecompany.com",
      "uid": "CA6ph0nTwdBzBwDg",
    }
  ]
}

```

```
        "name": "Mary Brown"
      }
    ]
  }
}
```

user.new

Adds a new user to the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "user.new",
  "params": [
    "account_id",
    "name",
    "email"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

account_id

Required string, The unique, system-generated account ID.

name

Required string, which can contain up to 64 characters. The name of the user.

email

Required string, which can contain up to 64 characters. The email address of the user.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "email": "string",
    "name": "string",
    "uid": "string"
  }
}
```

Response parameters

The result object contains the following key-value pairs:

email

String. The email address of the user.

name

String. The name of the user.

uid

String. The unique, system-generated user ID.

Permissions

The user must have administrator permissions for the account to which the specified user is associated.

Example

The following JSON-RPC request and response describe a successful call to the **user.new** method.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "user.new",
  "params": [
    "QrOXxEE9-fATtgAD",
    "TestUser",
    "someone@somewhere.com"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "email": "someone@somewhere.com",
    "uid": "Cmymc1NOpPMv5ADn",
    "name": "TestUser"
  }
}
```

See also

- >> ["user.permissions.set" on page 111](#)
- >> ["group.user.permission.add" on page 119](#)
- >> [group.user.permission.admin.set](#)

user.passwd.set

Sets the password of the specified user.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "user.passwd.set",
  "params": [
    "uid",
    "pwd"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

pwd

Required string. The new password of the specified user.

uid

Required string, which can contain exactly 16 characters. The unique, system-generated user ID of the specified user.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified user is associated.

Example

The following JSON-RPC request and response describe a successful call to the **user.passwd.set** method. The request sets the password for the specified user to "VeryL0ngU5erPassw0rd".

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "user.passwd.set",
  "params": [
    "Cmymc1NOpPMv5ADn",
    "VeryL0ngU5erPassw0rd"
  ]
}
```

```
]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

user.permissions.get

Get the permissions for a user for an account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "user.permissions.get",
  "params": [
    uid
  ]
}
```

Request parameters

Note: The following parameters are optional. If any parameter is omitted, that permission will not be changed for the specified user.

uid

Required string. The unique, system-generated user ID.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "account_admin": true or false,
    "client_admin": true or false,
    "client_viewer": true or false,
    "add_campaign": true or false
  }
}
```

Response parameters

account_admin

Required boolean. **true** if the specified user is an account administrator for its account; otherwise, **false**.

client_admin

Required boolean. **true** if the specified user is a account administrator for its account and all subaccounts; otherwise, **false**.

client_viewer

Required boolean. **true** can view all subaccounts of the account; otherwise, **false**.

add_campaign

Required boolean. **true** if the specified user can create campaigns for any account where they are also an admin; otherwise, **false**.

Permissions

The user must have administrator permissions for the account to which the specified user is associated.

Example

The following JSON-RPC request and response describe a successful call to the **user.permissions.get** method. The request gets user permissions for whether they are an account admin, an admin of client accounts, can view client accounts, or add campaigns.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "user.permissions.get",
  "params": [
    "Cmymc1NOpPMv5ADn"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "account_admin": false,
    "client_admin": false,
    "client_viewer": true,
    "add_campaign": true
  }
}
```

user.permissions.set

Set the permissions for a user for an account.

Request syntax

Required parameter is in bold.

```
{
  "jsonrpc": "2.0",
```

```

    "id": 1,
    "method": "user.permissions.set",
    "params": [
      {
        "uid": "system generated user id",
        "account_admin": true or false,
        "client_admin": true or false,
        "client_viewer": true or false,
        "add_campaign": true or false
      }
    ]
  }
}

```

Request parameters

The *params* array requires one (and only one) object with the following key-value pairs:

Note: The parameters regarding user permissions are optional. If any permission is omitted, that permission will not be changed for the specified user.

uid

Required string. The unique, system-generated user ID.

account_admin

Optional boolean. **true** if the specified user is an account administrator for its account; otherwise, **false**.

client_admin

Optional boolean. **true** if the specified user is a account administrator for its account and all subaccounts; otherwise, **false**.

client_viewer

Optional boolean. **true** can view all subaccounts of the account; otherwise, **false**.

add_campaign

Optional boolean. **true** if the specified user can create campaigns for any account where they are also an admin; otherwise, **false**.

Response syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}

```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified user is associated.

Example

The following JSON-RPC request and response describe a successful call to the **user.permissions.set** method. The request grants the user admissions on its account, permission to add campaigns on its account, and view subaccounts, but does not grant the user admin permissions on client accounts.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "user.permissions.set",
  "params": [
    {
      "uid": "Cnymc1NOpPMv5ADn",
      "account_admin": true,
      "client_admin": false,
      "client_viewer": true,
      "add_campaign": true
    }
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

Ad Groups

This section describes the methods available to manage ad groups.

Method	Description
<u>group.list</u>	Lists the ad groups associated with the specified account.
<u>group.new</u>	Adds a new ad group associated with the specified account.
<u>group.search</u>	Searches for ad groups in the specified account by the group name and/or description that you provide.
<u>group.user.permission.add</u>	Adds the specified user to the specified ad group.
<u>grp.descr.get</u>	Gets the description of the specified ad group.
<u>grp.descr.set</u>	Sets the description of the specified ad group.
<u>grp.name.get</u>	Gets the name of the specified ad group.

Method	Description
<u>grp.name.set</u>	Sets the name of the specified ad group.

group.list

Lists the active ad groups associated with the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "group.list",
  "params": [
    "account_id"
  ]
}
```

Request parameters

account_id

Required string. The unique, system-generated ID of the account that the groups belong to.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "descr": "string",
      "grpId": "string",
      "name": "string"
    }
  ]
}
```

Response parameters

The result is an array that contains 0 or more objects. Each object contains the following key-value pairs:

grpId

The unique, system-generated ID of the ad group.

name

The name of the ad group.

descr

The description of the ad group. An empty string ("") is returned if the group does not have a description.

Permissions

The method returns only those groups that the API user is a member of. Note that users with Account Admin permissions on the specified account are members of all groups in the account.

Example

The following JSON-RPC request and response describe a successful call to the **group.list** method. The response returns information about the ad groups associated with the account specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "group.list",
  "params": [
    "QrOXxEE9-fATtgAD"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "grpid": "fwABAVEkGEtLjQBY",
      "name": "Test Group 1",
      "descr": "Test group"
    },
    {
      "grpid": "QrOX60VTorYkdQCX",
      "name": "Test Group 2",
      "descr": "Test group"
    }
  ]
}
```

group.new

Adds a new ad group associated with the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "group.new",
  "params": [
    "acc",
    "nam",
    "dsc"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

acc	Required string. The unique, system-generated account ID of the specified account.
dsc	Required string. The description of the ad group.
nam	Required string. The name of the ad group.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "descr",
    "grpId",
    "name"
  }
}
```

Response parameters

descr	Optional string. The description of the ad group.
grpId	Optional string. The unique, system-generated group ID of the ad group.
name	Optional string. The name of the ad group.
result	Required object. An object that represents the new ad group associated with the specified account.

Permissions

The user must have user permissions for the specified account.

Example

The following JSON-RPC request and response describe a successful call to the **group.ad** method. The request defines a new ad group for the specified account, and the response returns information about the new ad group.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "group.new",
  "params": [
    "QrOXxEE9-fATtgAD",
    "Test Group 3",
    "Test group"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "grpId": "fwABAVEkGETLjQBZ",
    "name": "Test Group 3",
    "descr": "Test group"
  }
}
```

group.search

Searches for ad groups in the specified account by the group name and/or description that you provide. If you do not specify a name or description in the request, all groups associated with the specified account are returned.

Request syntax

Required parameter is in bold.

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "group.search",
  "params": [
    {
      "account_id": "string",
      "name": "string",
      "description": "string"
      "include_subaccts": boolean
    }
  ]
}
```

Request parameters

The *params* array requires one (and only one) object with the following key-value pairs:

account_id

Required string. The unique, system-generated account ID of the account that the groups belong to.

include_subaccts

Optional boolean. If **true**, groups belonging to the specified account and for all of its client accounts are returned. If not included or **false**, only groups belonging to the specified account are returned.

name

Optional string. If specified, the method does a case-insensitive search for group names that match the string you provide. If you include the description parameter in the request, both the specified name and description must match.

description

Optional string. If specified, the method does a case-insensitive search for group descriptions that match the string you provide. If you include the name parameter in the request, both the specified name and description must match.

You can use the following wildcard characters in the *name* and *description* search strings both before and after the other characters in the string:

- » The percent sign ("%") is a substitute for 0 or more characters.
- » The underscore ("_") is a substitute for exactly 1 character.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "account_id": "string",
      "group_id": "string",
      "name": "string",
      "description": "string"
    }
  ]
}
```

Response parameters

The result is an array that contains 0 or more objects, each of which contains the following key-value pairs:

account_id

The unique, system-generated account ID of the account that the groups belong to.

group_id

The unique, system-generated group ID of the ad group.

name

The name of the ad group.

descr

The description of the ad group. An empty string ("") is returned if the group does not have a description.

Permissions

The method returns all groups that the API user is a member of. Note that users with Account Admin permissions on the specified account are members of all groups in the account, and users with Client Admin permissions are members of all groups in all sub-accounts.

Remarks

As a best practice, if you do not want to specify search criteria, omit the search parameter (or parameters) in the request. Results may vary if you include a search parameter and set it to an empty string, null, or undefined.

Example

In the following example, the method searches for groups in the specified account that match the string in the *name* parameter.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
```

```

    "method": "group.search",
    "params": [
      {
        "account_id": "QrOXxEE9-fATtgAD",
        "name": "%google%"
      }
    ]
  }

```

Response

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "account_id": "QrOXxEE9-fATtgAD",
      "group_id": "fwABAVEkGETLjQBY",
      "name": "PPC: Google NYC",
      "description": "PPC Google in NYC area"
    },
    {
      "account_id": "QrOXxEE9-fATtgAD",
      "group_id": "QrOX60VTorYkdQCX",
      "name": "Google Paid Ads",
      "description": ""
    }
  ]
}

```

group.user.permission.add

Adds the specified user to the specified ad group.

Request syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "group.user.permission.add",
  "params": [
    "grp",
    "user"
  ]
}

```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

grp

Required string, which can contain exactly 16 characters. The unique, system-generated group ID of the specified ad group.

user

Required string, which can contain exactly 16 characters. The unique, system-generated user ID of the specified user.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have group administrator permissions for the specified ad group.

Example

The following JSON-RPC request and response describe a successful call to the **group.user.permission.add** method. The request adds the specified user to the membership of the specified ad group.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "group.user.permission.add",
  "params": [
    "CA6p1lAAOtFR8QAY",
    "CmymD1NNgSxFmWD-"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 26
}
```

grp.descr.get

Gets the description of the specified ad group.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "grp.descr.get",
  "params": [
    "grp"
  ]
}
```

Request parameters

grp

Required string, which can contain exactly 16 characters. The unique, system-generated group ID of the specified ad group.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "result"
}
```

Response parameters

result

Required string. A string that contains the description of the specified ad group.

Permissions

The user must have user permissions for the specified ad group.

Example

The following JSON-RPC request and response describe a successful call to the **grp.descr.get** method. The response returns the description of the ad group specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "grp.descr.get",
  "params": [
    "CA6p1lAAOtFR8QAY"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "Test group"}
```

grp.descr.set

Sets the description of the specified ad group.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "grp.descr.set",
  "params": [
    "grp",
    "descr"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

descr

Required string, which can contain up to 64 characters. The description of the specified ad group.

grp

Required string, which can contain exactly 16 characters. The unique, system-generated group ID of the specified ad group.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have group administrator permissions for the specified ad group.

Example

The following JSON-RPC request and response describe a successful call to the **grp.descr.set** method. The request sets the description of the specified ad group.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "grp.descr.set",
  "params": [
    "CA6p1lAAOtFR8QAY",
    "Test group"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

grp.name.get

Gets the name of the specified ad group.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "grp.name.get",
  "params": [
    "group_id"
  ]
}
```

Request parameters

group_id

Required string. The unique, system-generated ad group ID.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "string"
}
```

Response parameters

result

A string that contains the name of the specified ad group.

Permissions

The API user must be a member of the specified group. Note that users with the Account Admin permission are members of all groups in an account, and users with the Client Admin permission are members of all groups in all client accounts.

Example

The following JSON-RPC request and response describe a successful call to the **grp.name.get** method. The response returns the name of the ad group specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "grp.name.get",
  "params": [
    "CA6p1lAAOtFR8QAY"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "Test Group 3"
}
```

grp.name.set

Sets the name for the specified group.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "grp.name.set",
  "params": [
    "group_id",
    "group_name"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

group_id

Required string. The unique, system-generated group ID of the specified ad group.

group_name

Required string, which can contain up to 64 characters. The name of the specified ad group.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must be a Group Admin of the group specified group. Note that users with the Account Admin permission are Group Admins of all groups in an account.

Example

The following JSON-RPC request and response describe a successful call to the **grp.name.set** method. The request sets the name of the specified ad group.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "grp.name.set",
  "params": [
    "CA6p1lAAOtFR8QAY",
    "Test Group 3"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

Ad Campaigns

The following methods get and set ad campaign settings.

Method	Description
"ad.askexten.get" on the next page	Gets whether an incoming caller to the specified ad campaign must dial an extension.
ad.askexten.set	Sets whether an incoming caller to the specified ad campaign must dial an extension.
ad.busn_hrs.get	Gets business hours and call forwarding behavior for the specified ad campaign and day of the week.
ad.busn_hrs.set	Sets business hours and call forwarding behavior for the specified ad campaign and day of the week.
ad.configure	Creates a new ad campaign using the specified campaign options for the specified account or, if a campaign ID is specified, updates the campaign options of the specified ad campaign for the specified account.
ad.custom.define	Defines a campaign custom field for the specified account.
ad.custom.get	Gets the value of the specified campaign custom field, for the specified account and ad campaign.
ad.custom.set	Sets the value of the specified campaign custom field, for the specified account and ad campaign.
ad.deactivate	Deactivates the specified ad campaign.
ad.descr.get	Gets the description of the specified ad campaign.
ad.descr.set	Sets the description of the specified ad campaign.
ad.dispo.del	Deletes the specified tag from the specified ad campaign.
ad.dispo.list	Lists the tags for the specified ad campaign.
ad.dispo.new	Adds a tag to the specified ad campaign.
ad.dynamic.new	Creates a new dynamic tracking ad campaign.
ad.good_user.set	Sets the user to which notifications of good calls are emailed for the specified ad campaign.
ad.list	Lists the ad campaigns associated with the specified ad group.
ad.list.all	Lists the ad campaigns associated with the specified parent account, optionally constrained by a specified search filter.
ad.missed_user.set	Sets the user to which missed calls are emailed for the specified ad campaign.

Method	Description
<u>ad.name.get</u>	Gets the name of the specified ad campaign.
<u>ad.name.set</u>	Sets the name of the specified ad campaign.
<u>ad.new</u>	Adds a new ad campaign.
<u>ad.settings.copy</u>	Copy the campaign settings from the specified source ad campaign to the specified destination ad campaign.
<u>ad.settings.get</u>	Gets the campaign settings for the specified ad campaign.
<u>ad.timezone.set</u>	Sets the time zone for the specified ad campaign.
<u>timezone.list</u>	Gets a list containing the names of valid time zones.
<u>ad.vmail_user.set</u>	Sets the user assigned to receive email notifications when an incoming call goes to voicemail for the specified ad campaign.

ad.askexten.get

Gets whether an incoming caller to the specified ad campaign must dial an extension.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.askexten.get",
  "params": [
    "cmpid"
  ]
}
```

Request parameters

cmpid

Required string, which can contain exactly 16 characters. The unique, system-generated campaign ID of the specified ad campaign.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required boolean. **true** if an incoming caller to the specified ad campaign must dial an extension; otherwise, **false**.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.askexten.get** method.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.askexten.get",
  "params": [
    "CtjSZ1G1CNZF0AAm"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": true
}
```

ad.askexten.set

Sets whether an incoming caller to the specified ad campaign must dial an extension.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.askexten.set",
  "params": [
    "cmpid",
    askextn
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

askextn

Required boolean. **true** if an incoming caller to the specified ad campaign must dial an extension; otherwise, **false**.

cmpid

Required string, which can contain exactly 16 characters. The unique, system-generated campaign ID of the specified ad campaign.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.askexten.set** method. The request specifies that an incoming caller must dial an extension for the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.askexten.set",
  "params": [
    "CtjSZlG1CNZF0AAm",
    true
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

ad.busrn_hrs.get

Gets business hours and call forwarding behavior for the specified ad campaign and day of the week.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.busrn_hrs.get",
  "params": [
    "cmpid",
    day
  ]
}
```

Request parameters

cmpid

Required string, which can contain exactly 16 characters. The unique, system-generated campaign ID of the specified ad campaign.

day

Required integer, which can contain a value between 0 and 6. The index number for the day of the week. The following table describes the values allowed for this parameter:

Value	Description
0	Sunday
1	Monday
2	Tuesday
3	Wednesday
4	Thursday
5	Friday
6	Saturday

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "anumode": "string",
    "end": "string",
    "start": "string"
  }
}
```

Response parameters

anumode

String. A string that indicates call forwarding behavior for the specified ad campaign and day of the week. The following table describes the values allowed for this property:

Value	Description
always	Business hours are ignored and calls are always forwarded.
never	business hours are ignored and calls are never forwarded.
asis	Business hours are not ignored and calls are forwarded according to business hours.

If a value is not specified, the default value is "asis".

end

Optional string, that represents a valid ISO-8601 . An ISO-8601 datetime value representing the end time of business hours for the specified ad campaign and day of the week. The date portion of the value, if present, is stripped from the value.

Note Time zone designators are ignored. The time zone for the account is instead used.

result

Required object. An object representing the business hours and call forwarding behavior for the specified ad campaign.

start

Optional string, that represents a valid ISO-8601 . An ISO-8601 datetime value representing the start time of business hours for the specified ad campaign and day of the week. The date portion of the value, if present, is stripped from the value. **Note:** Time zone designators are ignored. The time zone for the account is instead used.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Examples

The following JSON-RPC request and response describe a successful call to the **ad.busrn_hrs.get** method. The response returns the business hours and forwarding mode for the specified day, Monday, and the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.busrn_hrs.get",
  "params": [
    "CtjSZ1G1CNZF0AAm",
    1
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "anumode": "asis",
    "end": "T17:00",
    "start": "T08:00"
  }
}
```

```
}  
}
```

ad.busrn_hrs.set

Sets business hours and call forwarding behavior for the specified ad campaign and day of the week.

Request syntax

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "method": "ad.busrn_hrs.set",  
  "params": [  
    "cmpid",  
    day,  
    "start",  
    "end",  
    "anumode"  
  ]  
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

cmpid

Required string, which can contain exactly 16 characters. The unique, system-generated campaign ID of the specified ad campaign.

day

Required integer, which can contain a value between 0 and 6. The index number for the day of the week. The following table describes the values allowed for this parameter:

Value	Description
0	Sunday
1	Monday
2	Tuesday
3	Wednesday
4	Thursday
5	Friday
6	Saturday

start

Required string, that represents a valid ISO-8601 time value for the start time of business hours for the specified ad campaign and day of the week.

end

Required string, that represents a valid ISO-8601 time value for the end time of business hours for the specified ad campaign and day of the week.

For both "start" and "end", the date portion of the value, if present, is ignored. The value for this property is specified using a 24-hour clock, in either "0000" or "00:00" formats. The value can be set from "00:00" to "23:59". The times are specified for the time zone of the ad campaign. For example, to set the end time to 6:00 PM, specify either "1800" or "18:00".

anumode

Required string. A string that indicates call forwarding behavior for the specified ad campaign and day of the week. The following table describes the values allowed for this property:

Value	Description
always	Open/Close hours are ignored and the full day is treated as During Business Hours
never	Open/Close hours are ignored and the full day is treated as After Business Hours
asis	Open / Close hours are not ignored and calls are forwarded according to the defined Open and Closed time for the day

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Examples

The following JSON-RPC requests and responses describe successful calls to the **ad.busrn_hrs.set** method.

Example 1: Forward during business hours

The request sets the business hours on Mondays to a typical range, from 8:00 AM to 5:00 PM, and forwards calls that occur inside that range based on business hours rules for the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.busrn_hrs.set",
  "params": [
    "CtjSZlG1CNZF0AAm",
    1,
    "T08:00",
    "T17:00",
    "asis"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

Example 2: Always forward

The request sets the business hours on Wednesdays to a typical range, from 8:00 AM to 5:00 PM, but overrides those rules to ensure that all calls, even those that occur outside that range, are forwarded for the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.busrn_hrs.set",
  "params": [
    "CtjSZlG1CNZF0AAm",
    3,
    "T08:00",
    "T17:00",
    "always"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

Example 3: Never forward

The request sets the business hours on Saturdays to a typical range, from 8:00 AM to 5:00 PM, but overrides those rules to ensure that all calls, even those that occur inside that range, are not forwarded for the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.busn_hrs.set",
  "params": [
    "CtjSZlG1CNZF0AAm",
    6,
    "T08:00",
    "T17:00",
    "never"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.configure

Creates a new ad campaign using the specified campaign options for the specified account or, if a campaign ID is included, updates the campaign options for that ad campaign. During creation, any values not explicitly set are assigned defaults consistent with new ad campaign creation in the Call Analytics user interface.

- » To assign an inbound number to the ad campaign, first provision a number for the account. Then provide that inbound number in the *inboundno* parameter when calling this method.
- » If an ad campaign requires playfiles, they must be uploaded to the account before you call this method.
- » Although not required, it is recommended that you set the time zone for the ad campaign. If not specified, the default value is "US/Eastern". See the *time_zone* parameter below for more information.

The parameters for **ad.configure** correspond to the ad campaign settings in the Call Analytics user interface.

Request syntax

Required parameters are in bold.

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.configure",
  "params": [
    "account_id",
    {

```

```

    "campaign_id": or "campaign_name": "string",
    "group_id": or "group_name": "string",
    "description": "test desc",
    "calldnaready_postback" : "string",
    "callmodelready_postback" : "string",
    "ask_extn": boolean,
    "call_postback": "string",
    "callerid": "string",
    "custom_code": "string",
    "description": "string",
    "enable_early_audio": boolean,
    "forw_timeout": integer,
    "good_user": "string",
    "inboundno": "string",
    "missed_user": "string",
    "precall_postback": "string",
    "record_calls": boolean,
    "redirect_url": "string",
    "shutdown_date": "string",
    "shutdown_dispo": "string",
    "time_zone": "string",
    "use_vmail": boolean,
    "vmail_user": "string",
    "winddown_number": "string",
    "winddown_start": "string",
    "business_hrs": [
      {
        "day_start": "hh:mm string",
        "day_end": "hh:mm string",
        "override": "string"
      }
    ],
    "forwards": [
      {
        "assignto": "string",
        "extension": "string",
        "hrsmode": "string",
        "matchrule": "string",
        "num_descr": "string",
        "number": "string",
        "status": "string",
        "step_order": integer,
        "timeout": integer,
        "weight": "string"
      }
    ],
    "playfiles": {
      "afterhourfile": "string",
      "aftrhr_exten_file": "string",
      "busnhr_exten_file": "string",

```



```

        "playfile": "string",
        "rejectfile": "string",
        "vmailfile": "string",
        "whisperfile": "string",
        "winddownfile": "string"
    }
}
]
}

```

Request parameters

account_id

Required string. The unique, system-generated account ID.

The ad campaign settings object contains the following key-value pairs:

campaign_id or *campaign_name*

Required string. Specify the *campaign_id* to update an existing campaign, or specify the *campaign_name* to create a new campaign. The *campaign_id* value is the unique, system-generated ID for the ad campaign. The *campaign_id* value is equivalent to the *cmpid* value used in other methods. The string for *campaign_name* can contain up to 64 characters.

group_id or *group_name*

Required string. The ad group to associate with the new or existing ad campaign. Specify either the system-generated group ID or the name (but not both) of an existing ad group, or specify the name of a new ad group that ad-configure will create. The string for *group_name* can contain up to 64 characters. When modifying an existing campaign, you must specify either the *group_id* or *group_name* even when you do not want to change the ad group that the campaign is associated with.

Note It is recommended that you use the *group_id* when you want to modify an existing campaign rather than the *group_name*. If you choose to specify the *group_name* of an existing ad group, the string must exactly match the name of the ad group (including case); otherwise, **ad.configure** will create a new ad group and associate the campaign with it.

description

Optional string. The description of the ad campaign.

calldnaready_postback

Optional string. Can contain up to 256 characters that represents a valid URI. The URL to which information about a call is posted back at the end of the call when Call DNA is available.

callmodelready_postback

Optional string. Can contain up to 256 characters that represents a valid URI. The URL to which information about a call is posted back at the end of the call when all call model outcomes are available.

ask_extn

Optional boolean. Specify **true** if the ad campaign plays an Ask Extension Playfile and allows the caller to enter an extension. If *ask_extn* is not specified, the default value is **false**.

call_postback

Optional string, which can contain up to 256 characters that represents a valid URI. The URL to which information about a call is posted back at the end of the call for the ad campaign.

callerid

Optional string, which can contain up to 20 characters. The caller ID string to display for the ad campaign.

custom_code

Optional string. The custom code, if desired, of the ad campaign.

enable_early_audio

Optional boolean. Indicates whether early audio will be attempted for calls to the ad campaign. If not specified, the default value is **false**.

forw_timeout

Optional integer. The time, in seconds, that Call Analytics has to forward calls to the specified forwarding phone number before ending the call (or sending it to Voicemail, if enabled). Adjust the value as needed to ensure callers get connected. The default value is 48 seconds.

good_user

Optional string. The unique, system-generated user ID for the user to which good calls to the campaign are emailed. The user must be in the same ad group as the campaign.

inboundno

Optional string. The inbound phone number (also referred to as the call tracking number) to use for the campaign. Specify a 10-digit phone number in a valid North American Numbering Plan (NANP) format, (for example: "8005551212"). You can use typical formats like dashes after the area code or parentheses around the area code (for example: "800-555-1212" or "(800)555-1212").

Note This value must be set to an available phone number in the number list for the parent account.

missed_user

Optional string. The unique, system-generated user ID for the user to which missed calls to the campaign are emailed. The user must be in the same ad group as the campaign.

precall_postback

Optional string, which can contain up to 256 characters that represents a valid URI. The URL to which information about a call is posted back at the start of the call for the ad campaign.

record_calls

Optional boolean. Specify **true** to enable call recording for the ad campaign. If not specified, the default value is **false**.

Note If this property is set to **true**, the ad campaign must include the *playfile* and *afterhourfile* in the [playfiles object](#).

redirect_url

Optional string, which can contain up to 256 characters that represents a valid URI. The URL to which a user is redirected after submitting a form for the ad campaign.

shutdown_date

Optional string, that represents a valid ISO 8601 datetime value; for example: "2014-12-10T00:00:00-8". The date on which the ad campaign will be deactivated.

shutdown_dispo

Optional string, which can contain up to 16 characters. The disposition of the inbound phone number, after deactivation, for the ad campaign.

The following table describes the values allowed for this property:

Value	Description
keep	Default. The phone number remains in the number pool for the account. However, the phone number cannot be used for new ad campaigns for one month after the deactivation date.
return	The phone number is returned, and is no longer available in the number pool for the account.

time_zone

Optional string, which can contain up to 32 characters. The name of a valid time zone. Use the ["timezone.list" on page 185](#) method to get a complete list of valid time zone names. Valid North American time zone names include: "US/Samoa", "US/Hawaii", "US/Aleutian", "US/Alaska", "US/Pacific", "US/Arizona", "US/Mountain",

"Canada/Saskatchewan", "US/Central", "US/Eastern", "America/Puerto_Rico", "Canada/Atlantic", and "Canada/Newfoundland". If not specified, the default value is "US/Eastern".

use_vmail

Optional boolean. **true** to enable Voicemail for the ad campaign; otherwise, **false**.

vmail_user

Optional string. The unique, system-generated user ID for the user to which Voicemail messages are emailed. The user must be in the same ad group as the campaign.

***business_hrs* array**

The *business_hrs* array is optional. If you set business hours, you can set different forwarding rules for calls that arrive when your business is open and when your business is closed. The *business_hrs* array contains an object that defines the business hours for one day. Typically, the array should contain seven objects to set the business hours for each day of the week. The first object sets the business hours for Sunday, and the seventh object sets the business hours for Saturday. The object contain the following key-value pairs:

day_start

Optional string. The start time of the business hours for a day of the week. The value for this property is specified using a 24-hour clock, in either "0000" or "00:00" formats. The value can be set from 00:00 to 23:59. For example, to set the start time to 8:00 AM, set this property to either "0800" or "08:00". Do not specify a time zone offset. (Use the *time_zone* key-value pair in the campaign settings object to set a time zone for the ad campaign.)

day_end

Optional string. The end time of the business hours for a day of the week. The value for this property is specified using a 24-hour clock, in either "0000" or "00:00" formats. The value can be set from 00:00 to 23:59. For example, to set the end time to 6:00 PM, set this property to either "1800" or "18:00". Do not specify a time zone offset. (Use the *time_zone* key-value pair in the campaign settings object to set a time zone for the ad campaign.)

override

Optional string, which can contain up to 16 characters. The override setting for a day of the week. If a value for this property is specified, the override setting overrides the business hours specified for this day of the week. To remove the override setting, do not specify a value for this property. The following table describes the values allowed for this property:

Value	Description
always	Open all day, regardless of the business hours for the day of the week.
never	Closed all day, regardless of the business hours of the week.

***forwards* array**

The *forwards* array is optional. The array contains an object for each call forwarding entry for the ad campaign. The object contains the following key-value pairs:

assignto

Optional string. The unique, system-generated user ID of the assigned user for the call forwarding entry of the ad campaign. The user must be in the same ad group as the campaign. If a value is not specified, the default value is "".

extension

Optional string, which can contain up to 32 characters. The extension number of the call forwarding entry for the ad campaign. The extension number can be preceded by one or more comma (,) characters, each of which represents a half-second pause, to allow enough time for the receiving phone to recognize the extension. If a value is not specified, the default value is "".

hrsmode

Optional string. Indicates when calls are forwarded by the call forwarding entry for the ad campaign. The following table lists the values allowed for this property:

Value	Description
busnhrs	Forward calls only during business hours
afterhrs	Forward calls only after business hours
both	Default. Always forward calls, regardless of business hours

matchrule

Optional string. An expression representing the advanced routing rules, if specified, for the call forwarding entry of the ad campaign. If a value is not specified, the default value is "".

num_descr

Optional string, which can contain up to 64 characters. The description of the call forwarding entry for the ad campaign. **Note:** If a value is not specified for this property, the user interface displays the values of the *number* and *extension* properties for the call forwarding entry. If a value is not specified, the default value is "".

number

Required string. The phone number of the call forwarding entry for the ad campaign. For the US and other countries in the North American Numbering Plan (NANP), this is a 10-digit phone number. When you supply a country code for a country not in the NANP (such as for Ireland), enter the phone number in E.164 format.

status

Optional string. Indicates the status of call forwarding entry for the ad campaign. The following table lists the values allowed for this property:

Value	Description
active	The call forwarding entry is active, and included when processing call forwarding entries.
inactive	The call forwarding entry is inactive, and excluded when processing call forwarding entries.

If a value is not specified, the default value is "active".

step_order

Optional integer, which can contain a value between 1 and 20. The call forwarding step of the call forwarding entry for the ad campaign. Call forwarding entries are processed in sequential order, in which all call forwarding entries at a given call forwarding step are evaluated before moving to the next call forwarding step. If a value is not specified, the default value is 1.

timeout

Optional integer. The time, in seconds, before the call forwarding time limit expires and ringing passes to the next call forwarding step (if any), or to Voicemail (if enabled), for the call forwarding entry of the ad campaign. If a value is not specified, the default value is 0.

weight

Optional string. The weighting of the call forwarding entry for the ad campaign. If more than one call forwarding entry is assigned to the same call forwarding step, calls are forwarded in proportion to the relative weight of each call forwarding entry at that call forwarding step. If a value is not specified, the default value is "0".

playfiles object

Following are the key-value pairs for setting playfiles for the ad campaign. You can specify either the system-generated file ID or the name of the playfile. If you specify a name, do not include the extension, and make sure the string exactly matches

the file name, including the case. **Note:** The playfiles must be uploaded to the account before you set them in the ad campaign.

afterhourfile

Optional string. The after-hours playfile for the ad campaign.

aftrhr_exten_file

Optional string. The after-hours ask extension playfile for the ad campaign.

busnhr_exten_file

Optional string. The business hours ask extension playfile for the ad campaign.

playfile

Optional string. The business hours playfile for the ad campaign.

rejectfile

Optional string. The call rejection playfile for the ad campaign.

vmailfile

Optional string. The voicemail playfile for the ad campaign.

whisperfile

Optional string. The whisper playfile for the ad campaign.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "string"
}
```

Response parameters

result

String. The unique, system-generated campaign ID of the new or existing ad campaign.

Permissions

The API user must have Account Admin and Allowed to Add Campaign permissions for the specified account.

Example

The following JSON-RPC request and response describe a successful call to the **ad.configure** method to update the business hours on an existing ad campaign. The request sets the business hours to be open all day on Sunday, open Monday and Tuesday from 6:30 AM to 6:00 PM, open Wednesday, Thursday, and Friday from 6:30 AM to 11:00 PM, and closed all day on Saturday.

Example Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.configure",
  "params": [
    "CmyMQVUInoMluAAu",
    {
      "campaign_id": "CmyMQVUInt0zvADV",
      "group_id": "CmyMQVUInoUluAAw",
    }
  ]
}
```

```

    "calldnaready_postback" : "http://mydomain123.com",
    "callmodelready_postback" : "http://mydomain.com",
    "forw_timeout": 50,
    "business_hrs": [
      {
        "override": "always"
      },
      {
        "day_start": "0630",
        "day_end": "1800"
      },
      {
        "day_start": "06:30",
        "day_end": "1800"
      },
      {
        "day_start": "0630",
        "day_end": "2300"
      },
      {
        "day_start": "0630",
        "day_end": "2300"
      },
      {
        "day_start": "0630",
        "day_end": "2300"
      },
      {
        "override": "never"
      }
    ]
  }
}

```

Example Response

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "string"
}

```

See also

» ["ad.dynamic.new" on page 160](#)

ad.custom.define

Defines a campaign custom field for the specified account. You use this method to define the name and data type of the custom field, and you use ["ad.custom.set" on page 149](#) to specify the value for the custom field.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.custom.define",
  "params": [
    "account_id",
    field_number,
    "field_name",
    "field_type",
    [
      "menu_value_1",
      "menu_value_2",
      ...,
      "menu_value_n"
    ]
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

account_id

Required string. The unique, system-generated account ID of the account that you want to configure.

field_number

Required integer, which can contain a value between 1 and 20. The index number of the custom field.

field_name

Required string, which can contain up to 32 characters. The name of the custom field. The name is used to identify the custom field. The name is displayed in the user interface, and you specify the name when programmatically setting the value of the field with the **ad.custom.set** method.

Note The custom field name must be unique across the entire account.

field_type

Required string. The type of the custom field. The custom field type restricts what values are allowed to be set for the custom field in the user interface and via **ad.custom.set**. The following table describes the supported data types:

Value	Description
bool	A Boolean value. A drop-down list containing "yes" and "no" is displayed in the user interface, from which a user can select a value.

Value	Description
date	A date value. A date picker is displayed in the user interface, from which a user can select a date value.
float	A floating-point decimal numeric value. A text box is displayed in the user interface, in which a user can provide a floating-point decimal value.
int	An integer numeric value. A text box is displayed in the user interface, in which a user can provide an integer numeric value.
menu	An array of strings that you specify in the <i>menu items array</i> . A drop-down list containing the strings specified in the array is displayed in the user interface, from which a user can select a value.
text	A free-form text value. A text box is displayed in the user interface, in which a user can provide a free-form text value.

menu items array

Required array. If *field_type* is anything other than **menu**, specify an empty array. For the **menu** *field_type*, specify an array of strings. Each string in the array is displayed in the user interface as an item in the menu's drop-down list. If you set a value for the menu via **ad.custom.set**, the string you specify in **ad.custom.set** must exactly match (including case) one of the strings that you have specified in this array.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

Accounts can define up to 20 campaign custom fields to support business-specific information for ad campaigns associated with that account.

Inheriting custom fields

By default, a client account inherits the campaign custom fields from its parent account, and you see the inherited fields in the user interface on the Campaign Settings page of each ad campaign in the client account. If you use **ad.-custom.define** or the user interface to define a campaign custom field for a client account, the client account no longer inherits campaign custom fields from the parent account, and only the custom field defined on the client account will be displayed in the Campaign Settings page of each ad campaign in the client account.

Examples

The following JSON-RPC requests and responses describe successful calls to the **ad.custom.define** method.

Example 1: Integer custom field

The following request sets the name of campaign custom field 1 for the specified account to **Campaign Size**, and configures the custom field to accept an integer value. The new name of the custom field is displayed in the user interface, which now accepts an integer value for the custom field.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.custom.define",
  "params": [
    "QrOXxEE9-fATtgAD",
    1,
    "Campaign Size",
    "int",
    []
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

Example 2: Menu custom field

The following request defines the name of campaign custom field 2 to **Campaign Region**, which is a menu. The name of the custom field is displayed in the user interface, and each string in the array is displayed as an item in the menu's drop-down list.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.custom.define",
  "params": [
    "QrOXxEE9-fATtgAD",
    2,
    "Campaign Region",
    "menu",
    [
      "Central",
      "East North Central",
      "Northeast",
    ]
  ]
}
```

```

        "Northwest",
        "South",
        "Southeast",
        "Southwest",
        "West",
        "West North Central"
    ]
}

```

Response

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}

```

ad.custom.get

Gets the value of the specified campaign custom fields for the specified account and ad campaign.

Request syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.custom.get",
  "params": [
    "campaign_id"
  ]
}

```

Request parameters

campaign_id

Required string. The unique, system-generated ID that identifies an campaign ID.

Response syntax

```

{
  "jsonrpc": "2.0",
  "id": "1",
  "result": [
    {
      "account_id": "QrOXy_XXB",
      "campaign_id": "CA6phXXQzb",
      "fields": {
        1 : {"label": "value"},

```

```

    2 : {"label":"value"},
    3 : {"label":"value"},
    4 : {"label":"value"},
    5 : {"label":"value"},
    6 : {"label":"value"},
    7 : {"label":"value"},
    8 : {"label":"value"},
    9 : {"label":"value"},
   10 : {"label":"value"},
   11 : {"label":"value"},
   12 : {"label":"value"},
   13 : {"label":"value"},
   14 : {"label":"value"},
   15 : {"label":"value"},
   16 : {"label":"value"},
   17 : {"label":"value"},
   18 : {"label":"value"},
   19 : {"label":"value"},
   20 : {"label":"value"}
  }
]
}

```

Response parameters

result

An object containing:
account_id

The unique, system-generated account ID

campaign_id

The unique, system-generated campaign ID

fields

An object containing the current labels and values for each of the 20 available campaign custom fields

that

follows the following format:
{ "label": "value" }

Permissions

The user must have administrator permissions for the specified account.

Example

Example Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.custom.get",
  "params": [
    "campaign_id"
  ]
}
```

Example Response

```
"jsonrpc": "2.0",
"id": 1,
"result": {
  "params": [
    "account_id": "QrOXxEE9",
    "campaign_id"
  ],
  "fields": [
    {
      "1": {
        "Date": ""
      },
      "2": {
        "MKWID" : ""
      },
      "3": {
        "offer"
      },
      "4": {
        "Campaign Type"
      },
      "5": {
        "Segmentation Name"
      },
      "6": {
        "name-campaign field"
      },
      "7": {
        "good?"
      },
      "8": {
        "cmp_custom8"
      },
      "9": {
        "datey date"
      },
      "10": {
        "cmp_custom10"
      },
      "11": {
```

```

        "Some float"
    },
    "12": {
        "new"
    },
    "13": {
        "Integration Test"
    },
    "14": {
        "menu test"
    },
    "15": {
        "cmp_custom15"
    },
    "16": {
        "cmp_custom16"
    },
    "17": {
        "cmp_custom17"
    },
    "18": {
        "cmp_custom18"
    },
    "19": {
        "cmp_custom19"
    },
    "20": {
        "Alphanumeric value"
    }
},
]
}

```

ad.custom.set

Sets the value of the specified campaign custom field for the specified account and ad campaign.

Request syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.custom.set",
  "params": [
    "account_id",
    "campaign_id",
    "field_name",

```

```

    "field_value"
  ]
}

```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

account_id

Required string. The unique, system-generated account ID of the account containing the campaign that you want to configure.

campaign_id

Required string. The unique, system-generated account ID of the campaign that you want to configure.

field_name

Required string. The name of the custom field to configure.

field_value

Required. The value of the campaign custom field. The value must be appropriate to the field type of the specified custom field, as described in the following table:

Field Type	Values
bool	<p>A string with one of the following values:</p> <ul style="list-style-type: none"> » "yes" or "1" » "no" or "0"
date	<p>A date string in the format "DD Mon YYYY" where:</p> <ul style="list-style-type: none"> » DD is a two-digit day. » Mon is the first three letters of the month. (Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec) » YYYY is the four-digit year <p>For example: "24 Mar 2016"</p>
float	<p>A floating-point decimal numeric value, for example: 3.14159</p> <p>Note you may also specify the value as a string, for example: "5.7"</p>
int	<p>An integer numeric value. For example: 155</p> <p>Negative integers are allowed. For example: -33</p> <p>Note you may also specify the value as a string, for example: "161"</p>
menu	<p>A string that matches exactly (including case) one of the strings defined for the menu.</p> <p>If you used "ad.custom.define" on page 143 to define the custom field, the value must be one of the strings that was specified in the <i>menu items array</i> request parameter.</p>
text	<p>A string with a maximum length of 64 characters.</p>

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

You must use the name of the campaign custom field to set its value. You cannot use the ordinal position of the custom field.

Example

The following JSON-RPC request and response describe a successful call to the **ad.custom.set** method. The request sets the value of the campaign custom field named **Campaign Size** to "20" for the specified account and ad campaign.

Example Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.custom.set",
  "params": [
    "QrOXxEE9-fATtgAD",
    "QrOXy0RPy_gATgEB",
    "Campaign Size",
    "20"
  ]
}
```

Example Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.deactivate

Deactivates the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.deactivate",
  "params": [
    "campaign_id"
  ]
}
```

Request parameters

The request contains the following parameter:

campaign_id

Required string. The unique, system-generated ID that identifies the ad campaign you want to deactivate.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must have Account Admin permissions in the account that the specified ad campaign is associated with.

Remarks

You can use **ad.deactivate** to deactivate [static](#) and [dynamic](#) ad campaigns. When you deactivate a static ad campaign, the [call tracking number](#) associated with the campaign is put in the [limbo](#) state. You can use ["number.return" on page 296](#) if you no longer need the number and won't be using it for another campaign.

Example

The following JSON-RPC request and response describe a successful call to the **ad.deactivate** method. The request deactivates the specified ad campaign.

Request


```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.deactivate",
  "params": [
    "QrOXy0RPY_gATgEB"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.descr.get

Gets the description of the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.descr.get",
  "params": [
    "cmpid"
  ]
}
```

Request parameters

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "result"
}
```

Response parameters

result

Required string. The description of the specified ad campaign.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.descr.get** method. The response returns the description, "Test campaign", of the ad campaign specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.descr.get",
  "params": [
    "QrOXy0RPy_gATgEB"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "Test campaign"
}
```

ad.descr.set

Sets the description of the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.descr.set",
  "params": [
    "cmpid",
    "descr"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

descr

Required string. The description of the specified ad campaign.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.descr.set** method. The request sets the description of the specified ad campaign to "Test campaign".

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.descr.set",
  "params": [
    "QrOxY0RPy_gATgEB",
    "Test campaign"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

ad.dispo.del

Deletes the specified tag from the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.dispo.del",

```

```
    "params": [
      "cmpid",
      "dispo"
    ]
  }
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

dispo

Required string, which can contain between 1 and 64 characters. The tag to delete from the specified ad campaign.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.dispo.del** method. The request deletes the specified tag, "Test Tag", from the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.dispo.del",
  "params": [
    "QrOXy0RPy_gATgEB",
    "Test Tag"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.dispo.list

Lists the tags for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.dispo.list",
  "params": [
    "campaign_id"
  ]
}
```

Request parameters

campaign_id

Required string. The unique, system-generated ID of the specified ad campaign. You can use ["ad.list" on page 169](#) or ["ad.list.all" on page 164](#) to get the campaign ID.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "dispo": "string"
    },
    {
      ...
    }
  ]
}
```

Response parameters

The result is an array of 0 or more objects, each of which contains the following key-value pair:

dispo

String. A tag for the specified ad campaign.

Permissions

The API user must belong to the same group as the specified ad campaign. Note that users with the Account Admin permission are members of all groups in an account, and users with the Client Admin permission are members of all groups in all client accounts.

Remarks

Users can select the campaign tags as they review calls in the call log in the Call Analytics user interface.

Example

The following JSON-RPC request and response describe a successful call to the **ad.dispo.list** method. The response returns an array of objects, each of which represents a tag for the ad campaign specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.dispo.list",
  "params": [
    "QrOXy0RPy_gATgEB"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "dispo": "Promising lead"
    },
    {
      "dispo": "Looky-loo"
    }
  ]
}
```

See also

» ["Ad Campaigns" on page 126](#)

ad.dispo.new

Adds a tag to the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.dispo.new",

```

```

    "params": [
      "cmpid",
      "dispo"
    ]
  }

```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

dispo

Required string, which can contain between 1 and 64 characters. The tag for the specified ad campaign

Response syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}

```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.dispo.new** method. The request adds a new tag, "Test Tag 1", to the specified ad campaign.

Request

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.dispo.new",
  "params": [
    "QrOXy0RPy_gATgEB",
    "Test Tag 1"
  ]
}

```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.dynamic.new

Creates a new dynamic tracking ad campaign with minimal settings.



Note: This method can be used only for accounts that have Keyword Tracking enabled. If this feature is not enabled on the specified account, an error occurs. To enable Keyword Tracking, contact Client Services.

Request syntax

Note Required parameters are in **bold**.

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.dynamic.new",
  "params": [
    "account_id",
    {
      "campaign_name": "string",
      "forward": "string",
      "group_id" or "group_name": "string",
      "preserve_case": boolean,
      "refdom": "string"
    }
  ]
}
```

Request parameters

The request contains the following parameters:

account_id

Required string. The unique, system-generated account ID.

The required object contains the following key-value pairs:

campaign_name

Required string. The name of the new dynamic tracking ad campaign. The specified string can contain up to 64 characters.

forward

Required string. The forwarding phone number to use for the new ad campaign. Although the forwarding phone number can be specified in the North American Numbering Plan (NANP) format consisting of a three-digit area code followed by a seven-digit local phone number, we encourage you to use the E.164 number format. E.164 is the standard format for international phone numbers, and it includes all the relevant information to route calls

worldwide. E.164 numbers have a maximum of fifteen digits and are prefixed with a "+" followed by the country code and then the subscriber's phone number including area code.

group_id or group_name

Optional string. Either the system-generated group ID or the name (but not both) for an existing ad group to which the new campaign will be added. If you do not specify either, the method searches for an ad group with the name "default" (which is the name of the default ad group unless it has been changed). Note that the method does a case sensitive search for the ad group.

preserve_case

Required boolean. Specify **true** to preserve keyword case; otherwise, specify **false**.

refdom

Optional string. The name of the referring domain for the dynamic tracking ad campaign. If a value is specified for this property, the dynamic tracking ad campaign is triggered only if a user comes from this domain; otherwise, the dynamic tracking ad campaign is triggered for all users. The specified string can contain up to 255 characters.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "string"
}
```

Response parameters

result

String. The unique, system-generated campaign ID of the new dynamic tracking ad campaign.

Permissions

The API user must have Account Admin and Allowed to Add Campaign permissions for the specified account.

Remarks

This method allows you to create a new dynamic tracking ad campaign using minimal settings. Once created, use the ["ad.configure" on page 135](#) method to provide more detailed settings, such as the time zone, business hours, and so on, for the new ad campaign.

Examples

The following JSON-RPC requests and responses describe successful calls to the **ad.dynamic.new** method.

Example 1

The response returns the campaign ID of the new dynamic tracking ad campaign defined for the account specified in the request. The request specifies a name and a forwarding phone number for the new dynamic tracking ad campaign, and indicates that the ad campaign does not preserve case for keywords. The ad campaign is added to the "default" ad group.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.dynamic.new",
  "params": [
    "QrOXxEEi-fATtgAD",
    {

```

```

        "campaign_name": "Basic KWLt campaign",
        "preserve_case": false,
        "forward": "2065550122"
    }
}

```

Response

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "CnymM1OGF74qxAET"
}

```

Example 2

The response returns the campaign ID of the new dynamic tracking ad campaign defined for the account specified in the request. The request specifies a referring domain and a forwarding phone number for the new dynamic tracking ad campaign, and indicates that the ad campaign preserves case for keywords. The ad campaign uses the specified domain, "myreferringdomain.com", as its referring domain, and the request assigns the ad campaign to the specified ad group. The forwarding phone number in the example is a London phone number in E.164 format. To convert the London phone number 020 7946 0591 to E.164, you start with the "+" prefix, add the UK country code (44), remove the leading 0 from the London dialing code (20), and append the rest of the local phone number (79460591).

Request

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.dynamic.new",
  "params": [
    "QrOXxEEi-fATtgAD",
    {
      "campaign_name": "Basic KWLt campaign",
      "preserve_case": true,
      "forward": "+442079460591",
      "refdom": "myreferringdomain.com",
      "groupi_id": "CnymM1Nn0IcqxDQ"
    }
  ]
}

```

Response

```

{
  "jsonrpc": "2.0",
  "id": 1,

```

```
}
  "result": "CmymM1OGF74qxAET"
}
```

ad.good_user.set

Sets the user to which notifications of good calls are emailed for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.good_user.set",
  "params": [
    "cmpid",
    "good_user"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

good_user

Required string, which can contain up to 16 characters. The unique, system-generated user ID for the user to which notifications of good calls are emailed. If a value is not specified, the default value is "".

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.good_user.set** method. The request indicates that notifications of good calls for the specified ad campaign should be sent to the email address of the specified user.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.good_user.set",
  "params": [
    "QrOXy0RPY_gATgEB",
    "CmymDlNNgSxFmWD-"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.list.all

Lists the ad campaigns associated with the specified account. If you specify a parent account, the accounts that are searched depend on the value of the *subacct* parameter. You can filter this list by supplying one or more of the optional search parameters described below. Although the search parameters are optional, an object is required even if it is empty.

Request syntax

Required parameters are in bold. The object is required even if it is empty.

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.list.all",
  "params": [
    "account id",

    {
      "page": integer,
      "status": "string",
      "subacct": boolean
    }
  ]
}
```

Request parameters

account id

Required string. The unique, system-generated account ID of the account to search. If you specify the account ID of a client account, you must either not include *subacct* in the request or set it to **false**.

The search parameters object allows you to filter the search. The object is required even if it is empty. The object contains the following optional key-value pairs:

page

Optional integer. The *page* parameter allows you to specify which page of results you want. If 0 is specified, or if *page* is not included in the request, all ad campaigns are returned.
If a positive value is specified, a page of at most 500 ad campaigns, at the specified offset index, is returned. If the specified value is greater than the number of pages in the results, an empty array is returned.
For example, the following table describes how paging is managed for a **ad.list.all** method call that can return 1400 ad campaigns:

Value	Description
0	Return all 1400 ad campaigns.
1	Return the first page of 500 ad campaigns, at positions 1 to 500.
2	Return the second page of 500 ad campaigns, at positions 501 to 1000.
3	Return the third and last page of 400 remaining ad campaigns, at positions 1001 to 1400.
4	Return an empty array.

status

Optional string. To filter the search results based on campaign status, enter one of the following strings: **active** or **disabled**. If *status* is not included in the request, all campaigns for the specified parent account are returned.

subacct

Optional boolean. **true** to search for campaigns in all client accounts of the specified parent account. Note, that the parent account is not searched when *subacct* is **true**. If not included or **false**, only campaigns for the specified account are searched. Note: If you set *subacct* to **true** when the specified account is a client account, the method returns an error.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "acc_name": "string",
      "acc_status": "string",
      "accid": "string",
      "cmp_custom1": "string",
      ...
      "cmp_custom20": "string",
      "cmp_name": "string",
      "cmp_status": "string",
      "cmpid": "string",
      "customid": "string",
      "descr": "string",
      "grp_name": "string",
      "grp_id": "string",
    }
  ]
}
```

```

        "inboundno": "string
      },
      {
        ...
      }
    ]
  }

```

Response parameters

The result is an array that contains 0 or more objects, each of which contains the following key-value pairs:

acc_name

String. The name of the account containing the ad campaign.

acc_status

String. The status of the account containing the ad campaign, either **active** or **disabled**.

accid

String. The unique, system-generated account ID of the account containing the ad campaign.

cmp_custom1 - cmp_custom20

String. The value of Custom Field *n* (where *n* is between 1 and 20) on the ad campaign.

customid

String. The custom account ID of the account containing the ad campaign. This is the Custom Ref. ID field from the Company Info and Settings page in the Call Analytics user interface.

cmp_name

String. The name of the ad campaign.

cmp_status

String. The status of the ad campaign, either **active** or **disabled**.

cmpid

String. The unique, system-generated ID of the ad campaign.

descr

String. The description of the ad campaign.

grp_name

String. The name of the group that ad campaign belongs to.

grp_id

String. The unique, system-generated group ID of ad group that ad campaign belongs to.

inboundno

String. The inbound telephone number used by the ad campaign. Also referred to as the [call tracking number](#).

Permissions

The required permissions depend on the value of *subacct*, as follows:

- » **true**: The API user must have Account Admin and Client Admin permissions on the specified parent account.
- » **false**: The API user must have Account Admin permissions in the specified account.

Remarks

To get the campaigns in both the parent account in all client accounts you have to call **ad.list.all** twice, once with *subacct* set to **false** to get the campaigns in the parent account, and again with *subacct* set to **true** to get the campaigns in all client accounts.

The following JSON-RPC requests and responses describe successful calls to the **ad.list.all** method.

Example 1: Basic

The response returns detailed information about all of the ad campaigns associated with the specified account, without any search parameters. The response was truncated to contain only two custom campaign fields for readability.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.list.all",
  "params": [
    "CA6plk5NvE8oYwC4",
    {}
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "cmp_custom1": "",
      ...
      "cmp_custom20": "",
      "cmp_status": "active",
      "cmpid": "CA6plk5NvE8oYwC6",
      "accid": "CA6plk5NvE8oYwC4",
      "grp_id": "CA6plk5NvE8oYwC5",
      "acc_status": "active",
      "inboundno": "8665550635",
      "acc_name": "Copy Pro National",
      "customid": "",
      "descr": "",
      "cmp_name": "National Campaign",
      "grp_name": "Print Directories"
    },
    {
      "cmp_custom1": "",
      ...
      "cmp_custom20": "",
      "cmp_status": "active",
      "cmpid": "CA6plU5NvQExLgAs",
      "accid": "CA6plk5NvE8oYwC4",
      "grp_id": "CA6plk5NvE8oYwC5",
      "acc_status": "active",
      "inboundno": "8665551234",
      "acc_name": "Copy Pro National",

```

```

        "customid": "",
        "descr": "",
        "cmp_name": "National Campaign",
        "grp_name": "Print Directories"
    }
]
}

```

Example 2: Filtering

The response returns detailed information about the active ad campaigns associated with the client accounts for the specified parent account. Note that campaigns in the parent account are not returned because `subacct` is **true**.

Request

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.list.all",
  "params": [
    "CA6plk5NvE8oYwC4",
    {
      "subacct": true,
      "status": "active"
    }
  ]
}

```

Response

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "cmp_custom1": "",
      ...
      "cmp_custom20": "",
      "cmp_status": "active",
      "cmpid": "CA6plk5NvE8ozAbc",
      "accid": "CA6plk5NvE8oBxyz",
      "grp_id": "CA6plk5NvE8oYwru",
      "acc_status": "active",
      "inboundno": "8665550635",
      "acc_name": "Copy Pro NE",
      "customid": "",
      "descr": "",
      "cmp_name": "Metro North, MA",
      "grp_name": "Print Directories"
    },
    {

```



```

        "cmp_custom1": "",
        ...
        "cmp_custom20": "",
        "cmp_status": "active",
        "cmpid": "CZ6plU5NvQEabcd",
        "accid": "CX6plk5NvE8oYw123",
        "grpId": "CA6plk5NvE8oYwC5",
        "acc_status": "active",
        "inboundno": "8665550992",
        "acc_name": "Copy Pro SE",
        "customid": "",
        "descr": "",
        "cmp_name": "Atlanta",
        "grp_name": "Print Directories"
    },
    {
        ...
    }
]
}

```

See also

» ["ad.list" below](#)

ad.list

Lists the ad campaigns associated with the specified ad group.

Request syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.list",
  "params": [
    "group_id"
  ]
}

```

Request parameters

group_id

Required string. The unique, system-generated ad group ID. You can use ["group.search" on page 117](#) to obtain the group ID.

Response syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,

```

```

    "result": [
      {
        "cmpid": "string",
        "descr": "string",
        "inboundno": "string",
        "name": "string"
      }
    ]
  }

```

Response parameters

The result is an array that contains 0 or more objects. Each object contains the following key-value pairs:

cmpid

Required string. The unique, system-generated campaign ID of the ad campaign.

descr

Required string. The description of the ad campaign.

inboundno

Required string. The inbound telephone number used by the ad campaign.

name

Required string, which can contain up to 64 characters. The name of the ad campaign.

Permissions

The API user must be a member of the specified group. Note that users with the Account Admin permission are members of all groups in an account, and users with the Client Admin permission are members of all groups in all client accounts.

Remarks

This method provides basic information about the ad campaigns associated with a given account. For more detailed information about ad campaigns, or to provide a filtered or paged list of ad campaigns, use the ["ad.list.all" on page 164](#) method.

Example

The following example shows a successful call to the **ad.list** method. The response returns an array of objects, each of which contains basic information about an ad campaign that is a member of the specified ad group.

Request

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.list",
  "params": [
    "CA6phEqC8slAmAIT"
  ]
}

```

Response

```

{
  "jsonrpc": "2.0",

```

```

    "id": 1,
    "result": [
      {
        "cmpid": "CA6phUqDKpxb-gBk",
        "inboundno": "8882744740",
        "name": "Test Campaign 1",
        "descr": ""
      },
      {
        "cmpid": "CA6phUqDKpRagBRl",
        "inboundno": "8882744741",
        "name": "Test Campaign 2",
        "descr": ""
      }
    ]
  }

```

See also

- » ["Ad Campaigns" on page 126](#)
- » ["ad.list.all" on page 164](#)

ad.missed_user.set

Sets the user to which missed calls are emailed for the specified ad campaign.

Request syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.missed_user.set",
  "params": [
    "cmpid",
    "missed_user"
  ]
}

```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

missed_user

Required string, which can contain exactly 16 characters. The unique, system-generated user ID for the user to which missed calls are emailed. If a value is not specified, the default value is "".

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.missed_user.set** method. The request indicates that notifications of unanswered calls for the specified ad campaign should be sent to the email address of the specified user.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.missed_user.set",
  "params": [
    "QrOXy0RPy_gATgEB",
    "CmymDlNNgSxFmwD-"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.name.get

Gets the name of the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.name.get",

```

```
    "params": [
      "campaign_id"
    ]
  }
```

Request parameters

campaign_id

Required string. The unique, system-generated campaign ID. You can use the ["ad.list.all"](#) on page 164 or ["ad.list"](#) on page 169 to obtain the campaign ID.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "string"
}
```

Response parameters

result

String. The name of the specified ad campaign.

Permissions

The API user must belong to the same group as the specified ad campaign. Note that users with the Account Admin permission are members of all groups in an account, and users with the Client Admin permission are members of all groups in all client accounts.

Example

The following example shows a successful call to the **ad.name.get** method. The response returns the name of the ad campaign specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.name.get",
  "params": [
    "QrOXy0RPY_gATgEB"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "Test Campaign"
}
```

ad.name.set

Sets the name of the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.name.set",
  "params": [
    "campaign_id",
    "campaign_name"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

campaign_id

Required string. The unique, system-generated campaign ID.

campaign_name

Required string, which can contain up to 64 characters. The name of the specified ad campaign.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must be a Group Admin of the group that the specified campaign belongs to. Note that users with the Account Admin permission are Group Admins of all groups in an account, and parent account users with the Client Admin permission are Group Admins of all groups in all sub-accounts.

Example

The following example shows a successful call to the **ad.name.set** method. The request sets the name of the specified ad campaign to "Test Campaign".

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.name.set",
  "params": [
    "QrOXy0RPY_gATgEB",
    "Test Campaign"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

ad.new

Creates a new [static](#) ad campaign in the account that the specified group belongs to.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.new",
  "params": [
    "group_id",
    "name",
    "description",
    "inbound_number"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

group_id

Required string. The unique, system-generated group ID of the ad group that the new ad campaign will belong to. You can use ["group.list" on page 114](#) or ["group.search" on page 117](#) to find the group ID that you want.

name

Required string. The name for the ad campaign.

description

Required string. The description for the ad campaign.

inbound_number

Required string. The inbound phone number (also referred to as the [call tracking number](#)) to use for the campaign. Specify a 10-digit phone number in a valid [North American Numbering Plan](#) (NANP) format, (for example: "8005551212"). The string must contain only numbers. Do not include hyphens or other characters. The specified phone number must match an available phone number in the number list of the parent account that the group is in. The method returns an error if the phone number is in use by another campaign or has the status of [limbo](#). You can use ["number.list" on page 274](#) or ["number.avail" on page 271](#) to find available numbers.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "cmpid": "string",
    "number": "string"
  }
}
```

Response parameters

The result object contains the following key-value pairs:

cmpid

The unique, system-generated campaign ID of the new ad campaign.

number

The inbound phone number (that was specified in the request) of the new ad campaign.

Permissions

The API user must have been created in a parent account. To create campaigns in a parent account, the API user must have Account Admin and Allowed to Add Campaign permissions. To create campaigns in a sub-account, the API user must have Account Admin, Client Admin, and Allowed to Add Campaign permissions.

Remarks

If you need to create campaigns in a sub-account with an API user that was created in the sub-account, use ["ad.configure" on page 135](#).

Examples

The following JSON-RPC requests and responses describe successful calls to the **ad.new** method.

Example 1: A static ad campaign

The response returns an object containing information about the new ad campaign specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.new",
  "params": [
    "CmyM1Nn0IcqxDQ",
    "TestCampaign",
    "Test campaign",
  ]
}
```



```
      "8005551212"
    ]
  }
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "cmpid": "CmymM1N_gHoqxAEQ",
    "number": "8005551212"
  }
}
```

ad.settings.copy

Copy the campaign settings from the specified source ad campaign to the specified destination ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.settings.copy",
  "params": [
    "src_cmpid",
    "dest_cmpid"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

`src_cmpid`

Required string, which can contain exactly 16 characters. The unique, system-generated campaign ID for the source ad campaign.

`dest_cmpid`

Required string, which can contain exactly 16 characters. The unique, system-generated campaign ID for the destination ad campaign.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have user permissions for the account to which the specified source ad campaign is associated, and group administrator permissions for the account to which the specified destination ad campaign is associated.

Remarks

Settings can only be copied between ad campaigns within the same account. This method does not check parent accounts, so settings cannot be copied between ad campaigns associated with different client accounts for the same parent account.

Note Both ad campaigns must be configured for campaign level settings.

Example

The following JSON-RPC request and response describe a successful call to the **ad.settings.copy** method. The settings of the specified source ad campaign are copied to the specified destination ad campaign, overwriting any existing values.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.settings.copy",
  "params": [
    "CA6phklnndJldgD1",
    "CA6phEtWMT47mgNN"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 0
}
```

ad.settings.get

Gets the campaign settings for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.settings.get",
  "params": [
    "campaign_id"
  ]
}
```

```
}
]
```

Request parameters

campaign_id

Required string. The unique, system-generated campaign ID. You can use the ["ad.list.all"](#) on page 164 or ["ad.list"](#) on page 169 to obtain the campaign ID.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "afterhrs": "string",
    "afterhr_exten": "string",
    "answerapp": "string",
    "answerappid": "string",
    "askexten": boolean,
    "busnhr_exten": "string",
    "c2c.postback": "string",
    "c2c.redirect": "string",
    "callerid": "string",
    "enable_early_audio": boolean,
    "error": "string",
    "forwtout" integer,
    "good_user": "string",
    "missed_user": "string",
    "play": "string",
    "recordcall": "string",
    "redirect": "string",
    "reject": "string",
    "timezone": "string",
    "usevmail": boolean,
    "vmail": "string",
    "vmail_user": "string",
    "whisper": "string",
  }
}
```

Response parameters

The *result* object contains the following key-value pairs.

afterhrs

String. The unique, system-generated file ID of the after hours playfile for the ad campaign.

afterhr_exten

String. The unique, system-generated file ID of the after hours ask extension playfile for the ad campaign.

askexten

Boolean. **true** if the ad campaign plays an Ask Extension Playfile and allows the caller to enter an extension; otherwise, **false**.

busnhr_exten
String. The unique, system-generated file ID of the business hours ask extension playfile for the ad campaign.

callerid
String, which can contain up to 20 characters. The caller ID string to display for the ad campaign.

enable_early_audio
Boolean. Indicates whether early audio will be attempted for calls to the ad campaign.

error
Note The error playfile is no longer used. The value of this property is "".

forwrtout
Optional integer. The global timeout, in seconds, before a call should be forwarded according to the call forwarding settings for the ad campaign. If a value is not specified, the default value is 36.

good_user
String. The unique, system-generated user ID for the user to which good calls are emailed.

missed_user
String. The unique, system-generated user ID for the user to which missed calls are emailed.

play
String. The unique, system-generated file ID of the business hours playfile for the ad campaign.

recordcall
Boolean. **true** to enable call recording for the ad campaign; otherwise, **false**.
Note If this property is set to **true**, the ad campaign must include *busnhr_exten* and *afterhrs* playfiles.

redirect
String, which can contain up to 256 characters that represents a valid URI. The URL to which a user is redirected after submitting a form for the ad campaign.

reject
String. The unique, system-generated file ID of the call rejection playfile for the ad campaign. I

timezone
String. The name of the time zone set on the campaign.

usevmail
Boolean. **true** to enable voicemail for the ad campaign; otherwise, **false**.

vmail
String. The unique, system-generated file ID of the voicemail playfile for the ad campaign.

vmail_user
String. The unique, system-generated user ID for the user to which voicemail messages are emailed.

whisper
String. The unique, system-generated file ID of the whisper playfile for the ad campaign.

Permissions

The API user must belong to the same group as the specified ad campaign. Note that users with the Account Admin permission are members of all groups in an account, and users with the Client Admin permission are members of all groups in all client accounts.

Remarks

Use the ["ad.configure" on page 135](#) method to edit multiple settings at once, or one of the following methods to edit a single specific setting for an ad campaign.

Setting	Method
afterhr_exten	ad.play.afterhr_exten.set

Setting	Method
afterhrs	ad.play.afterhrs.set
askexten	ad.askexten.set
busnhr_exten	ad.play.busnhr_exten.set
c2c.postback	ad.c2c.postback.set
c2c.redirect	ad.c2c.redirect.set
callerid	ad.callerid.set
enable_early_audio	ad.enable_early_audio.set
error	ad.play.error.set
forwtout	ad.forwtout.set
good_user	ad.good_user.set
missed_user	ad.missed_user.set
play	ad.play.set
recordcall	ad.recordcall.set
redirect	ad.redirect.set
timezone	ad.timezone.set
usevmail	ad.usevmail.set
vmail	ad.play.vmail.set
vmail_user	ad.vmail_user.set
whisper	ad.play.whisper.set

Example

The following JSON-RPC request and response describe a successful call to the **ad.settings.get** method. The response returns the settings for the ad campaign specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.settings.get",
  "params": [
    "QrOXy0RPy_gATgEB"
  ]
}
```

```
}
]
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "missed_user": "CmymD1NNgSxFmWD-",
    "afterhr_exten": "",
    "answerappid": "",
    "c2c.postback": "http://mydomain.com/c2c_postback",
    "vmail": "",
    "recordcall": false,
    "callerid": "2063314444",
    "enable_early_audio": false,
    "redirect": "http://mydomain.com/redirect.php",
    "forwtout": "",
    "busnhr_exten": "",
    "error": "",
    "afterhrs": "",
    "vmail_user": "",
    "answerapp": false,
    "timezone": "US/Pacific",
    "play": "",
    "good_user": "CmymD1NNgSxFmWD-",
    "usevmail": true,
    "c2c.redirect": "http://mydomain.com/c2c_redirect.php",
    "wrapup": false,
    "whisper": "",
    "reject": "",
    "askexten": true,
    "wrapupid": "QrOXy0LX3ZgMCgAy"
  }
}
```

See also

- » ["ad.settings.copy" on page 177](#)
- » ["Ad Campaigns" on page 126](#)

ad.timezone.set

Sets the time zone for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
```

```

    "id": 1,
    "method": "ad.timezone.set",
    "params": [
        "campaign_id",
        "time_zone"
    ]
}

```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

campaign_id

Required string. The unique, system-generated campaign ID. You can use ["ad.list" on page 169](#) or ["ad.list.all" on page 164](#) to get the *campaign_id*.

time_zone

Required string containing the name of a valid time zone to set on the specified ad campaign. You can use ["timezone.list" on page 185](#) to get a list of time zones.

Response syntax

```

{
    "jsonrpc": "2.0",
    "id": 1,
    "result": integer
}

```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must be a Group Admin of the group that the specified campaign belongs to. Note that users with the Account Admin permission are Group Admins of all groups in an account, and parent account users with the Client Admin permission are Group Admins of all groups in all sub-accounts.

Remarks

The time zone controls important features, such as the business hours, call forwarding schedules, and call log entry times, for the ad campaign. Use the **timezone.list** method to get a list of valid Greenwich Mean Time (GMT) time zones that can be used with this method.

Example

The following JSON-RPC request and response show a successful call to the **ad.timezone.set** method. The request sets the time zone for the specified ad campaign to the "Europe/Paris" GMT time zone.

Request

```

{
    "jsonrpc": "2.0",

```

```
{
  "id": 1,
  "method": "ad.timezone.set",
  "params": [
    "QrOXy0RPy_gATgEB",
    "Europe/Paris"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.vmail_user.set

Sets the user assigned to receive email notifications when an incoming call goes to voicemail for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.vmail_user.set",
  "params": [
    "cmpid",
    "vmail_user"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

`cmpid`

Required string. The unique, system-generated campaign ID of the specified ad campaign.

`vmail_user`

Required string, which can contain exactly 16 characters. The unique, system-generated user ID for the user assigned to receive email notifications when an incoming call goes to voicemail. If a value is not specified, the default value is "".

Response syntax


```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Remarks

The voicemail user receives an email containing the recording of a voicemail message from an incoming call for the specified ad campaign, if voicemail is enabled for that ad campaign. To enable or disable voicemail for an ad campaign, use the **ad.usevmail.set** method. To get the voicemail user for an ad campaign, use the **ad.settings.get** method.

Example

The following JSON-RPC request and response describe a successful call to the **ad.vmail_user.set** method. The request sets the voicemail user for the specified ad campaign to the specified user.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.vmail_user.set",
  "params": [
    "QrOXy0RPy_gATgEB",
    "Cmymc1NNy2Av0gBl"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

timezone.list

Gets a list containing the names of valid time zones.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "timezone.list",
  "params": [
  ]
}
```

Request parameters

This method does not have any request parameters.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    "Africa/Abidjan", "Africa/Accra", "Africa/Addis_Ababa",
    ...,
    "US/Alaska", "US/Aleutian", "US/Arizona", "US/Central",
    "US/East-Indiana", "US/Eastern", "US/Hawaii", "US/Indiana-Starke",
    "US/Michigan", "US/Mountain", "US/Pacific", "US/Samoa",
    ...
  ]
}
```

Response parameters

result

An array of strings, each of which represents the name of a valid time zone.

Permissions

None. The API user just needs valid credentials.

Remarks

Use this method to get a list of valid Greenwich Mean Time (GMT) time zones that can be used with other method calls, such as "[ad.timezone.set](#)" on page 182.

Call Forwarding

The following methods get and set call forwarding settings on ad campaigns

Method	Description
ad.forw.assignto.set	Sets the user assigned to receive incoming calls for the specified call forwarding entry and ad campaign.
ad.forw.del	Deletes the call forwarding entry for a telephone number.

Method	Description
<u>ad.forw.exten.set</u>	Sets the extension number to which incoming calls are forwarded for the specified call forwarding entry and ad campaign.
<u>ad.forw.get</u>	Gets the call forwarding entry for the specified telephone number and ad campaign.
<u>ad.forw.hrsmode.set</u>	Sets when calls are forwarded by the call forwarding entry for the specified ad campaign.
<u>ad.forw.list</u>	Lists the call forwarding entries for the specified ad campaign.
<u>ad.forw.matchrule.set</u>	Sets an expression representing the advanced routing rules for the specified call forwarding entry and ad campaign.
<u>ad.forw.new</u>	Creates a new call forwarding entry that forwards inbound calls to the specified telephone number, for the specified ad campaign.
<u>ad.forw.status.set</u>	Sets the status of the specified call forwarding entry for the specified ad campaign.
<u>ad.forw.step.set</u>	Sets the call forwarding step of the specified call forwarding entry, for the specified ad campaign.
<u>ad.forw.timeout.set</u>	Sets the time, in seconds, before the call forwarding time limit expires and ringing passes to the next call forwarding step (if any), or to voicemail (if enabled), of the specified call forwarding entry for the specified ad campaign.
<u>ad.forw.weight.set</u>	Sets the weighting of the specified call forwarding entry for the specified ad campaign.
<u>ad.forwtout.get</u>	Gets the global call forwarding timeout for the specified ad campaign.
<u>ad.forwtout.set</u>	Sets the global call forwarding timeout for the specified ad campaign.

ad.forw.assignto.set

Sets the user assigned to receive incoming calls for the specified call forwarding entry and ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.assignto.set",
  "params": [
    "cmpid",
    "number",
    "uid"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

number

Required string, which can contain exactly 10 characters. The telephone number of the call forwarding entry to which the specified user is assigned.

Note The specified telephone number must match one of the telephone numbers in the call forwarding settings for the specified ad campaign; otherwise, an error occurs.

uid

Required string. The unique, system-generated user ID of the specified user.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.forw.assignto.set** method. The request assigns all incoming calls from the specified telephone number to the specified user for the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.assignto.set",
  "params": [
    "CA6phEtWMT47mgNN",
    "2064125430",
    "CmymDlNNgSxFmwd-"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.forw.del

Deletes the call forwarding entry for a telephone number.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.del",
  "params": [
    "cmpid",
    "number"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

number

Required string, which can contain exactly 10 characters. The telephone number of the call forwarding entry to be deleted.

Note The specified telephone number must match one of the telephone numbers in the call forwarding settings for the specified ad campaign; otherwise, an error occurs.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Remarks

To update a call forwarding entry, first use this method to delete the existing call forwarding entry, and then use the **ad.forw.new** method to replace it with the updated information.

Example

The following JSON-RPC request and response describe a successful call to the **ad.forw.del** method. The request deletes the call forwarding entry for the specified number from the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.del",
  "params": [
    "CA6phEtWMT47mgNN",
    "2064125430"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result":
}
```

ad.forw.exten.set

Sets the extension number to which incoming calls are forwarded for the specified call forwarding entry and ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.exten.set",
  "params": [
    "cmpid",
    "number",
    "extension"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

number

Required string, which can contain exactly 10 characters. The telephone number of the specified call forwarding entry.

Note The specified telephone number must match one of the telephone numbers in the call forwarding settings for the specified ad campaign; otherwise, an error occurs.

extension

Required string, which can contain up to 32 characters. The extension number to which incoming calls are forwarded for the specified call forwarding entry and ad campaign.

The extension number can be preceded by one or more comma (,) characters, each of which represents a half-second pause, to allow enough time for the receiving telephone to recognize the extension. If a value is not specified, the default value is "".

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.forw.exten.set** method. The request sets the extension of the specified forwarding number for the specified account to "0105", with a 2-second delay to allow enough time for the receiving telephone to recognize the extension.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.exten.set",
  "params": [
    "CA6phEtWMT47mgNN",
    "2064125430",
    ",,,0105"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
```

```
{
  "id": 1,
  "result": 1
}
```

ad.forw.get

Gets the call forwarding entry for the specified telephone number and ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.get",
  "params": [
    "cmpid",
    "number"
  ]
}
```

Request parameters

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

number

Required string, which can contain exactly 10 characters. The telephone number of the call forwarding entry.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "assignto",
    "descr",
    "extension",
    "hrsmode",
    "matchrule",
    "number",
    "status",
    step,
    extension,
    "weight"
  ]
}
```

Response parameters

assignto

Optional string, which can contain up to 16 characters. The unique, system-generated user ID of the assigned user for the call forwarding entry of the ad campaign. If a value is not specified, the default value is "".

descr
Optional string, which can contain up to 64 characters. The description of the call forwarding entry for the ad campaign.

Note If a value is not specified for this property, the user interface displays the values of the `number` and `extension` properties for the call forwarding entry. If a value is not specified, the default value is "".

extension
Optional integer. The time, in seconds, before the call forwarding time limit expires and ringing passes to the next call forwarding step (if any), or to voicemail (if enabled), for the call forwarding entry of the ad campaign. If a value is not specified, the default value is 0.

hrsmode
Optional string. Indicates when calls are forwarded by the call forwarding entry for the ad campaign. The following table lists the values allowed for this property:

Value	Description
busnhrs	Forward calls only during business hours
afterhrs	Forward calls only after business hours
both	Default. Always forward calls, regardless of business hours

If a value is not specified, the default value is "both".

matchrule
Optional string. An expression representing the advanced routing rules, if specified, for the call forwarding entry of the ad campaign. If a value is not specified, the default value is "".

number
Required string, which can contain up to 32 characters. The telephone number of the call forwarding entry for the ad campaign.

result
Required object. An object that represents the call forwarding entry for the specified telephone number and ad campaign.

status
Optional string. Indicates the status of call forwarding entry for the ad campaign. The following table lists the values allowed for this property:

Value	Description
active	The call forwarding entry is active, and included when processing call forwarding entries.
inactive	The call forwarding entry is inactive, and excluded when processing call forwarding entries.

If a value is not specified, the default value is "active".

step
Optional integer, which can contain a value between 1 and 20. The call forwarding step of the call forwarding entry for the ad campaign. Call forwarding entries are processed in sequential order, in which all call forwarding entries at a given call forwarding step are evaluated before moving to the next call forwarding step. If a value is not specified, the default value is 1.

weight

Optional string. The weighting of the call forwarding entry for the ad campaign. If more than one call forwarding entry is assigned to the same call forwarding step, calls are forwarded in proportion to the relative weight of each call forwarding entry at that call forwarding step. If a value is not specified, the default value is "0".

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.forw.get** method. The response returns information about the call forwarding entry for the telephone number and account specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.get",
  "params": [
    "CmymMlNZcDoqwwA-",
    "2062370002"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "hrsmode": "busnhrs",
    "extension": "0005",
    "number": "2062370002",
    "status": "active",
    "assignto": "CmymDlNNgSxFmwd-",
    "descr": "Test description",
    "step": 1,
    "matchrule": "PA & (215 + 267)",
    "timeout": 60,
    "weight": 0
  }
}
```

ad.forw.hrsmode.set

Sets when calls are forwarded by the call forwarding entry for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
```

```

    "id": 1,
    "method": "ad.forw.hrsmode.set",
    "params": [
        "cmpid",
        "number",
        "hrsmode"
    ]
}

```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

number

Required string, which can contain exactly 10 characters. The telephone number of the call forwarding entry for the specified ad campaign.

hrsmode

Required string, which can contain up to 32 characters. Indicates when calls are forwarded by the call forwarding entry for the ad campaign. The following table lists the values allowed for this property:

Value	Description
busnhrs	Forward calls only during business hours
afterhrs	Forward calls only after business hours
both	Default. Always forward calls, regardless of business hours

If a value is not specified, the default value is "both".

Response syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}

```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.forw.hrsmode.set** method. The request sets the call forwarding entry for the specified telephone number and ad campaign to forward calls only during business hours.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.hrsmode.set",
  "params": [
    "CA6phEtWMT47mgNN",
    "2064125430",
    "busnhrs"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

ad.forw.list

Lists the call forwarding entries for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.list",
  "params": [
    "campaign_id"
  ]
}
```

Request parameters

campaign_id

Required string. The unique, system-generated campaign ID. You can use ["ad.list" on page 169](#) or ["ad.list.all" on page 164](#) to find the campaign ID.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
```

```

    "result": [
      result: {
        "assignto": "string",
        "descr": "string",
        "extension": "string",
        "hrsmode": "string",
        "number": "string",
        "status": "string",
        "step": integer
      }
    ]
  }
}

```

Response parameters

The result is an array that contains 0 or more objects. Each object contains the following key-value pairs:

assignto

String. The unique, system-generated user ID of the assigned user for the call forwarding entry of the ad campaign. If a user has not been assigned to the call forwarding entry, an empty string ("") is returned.

descr

String, which can contain up to 64 characters. The description of the call forwarding entry for the ad campaign. If a value is not specified, an empty string is returned.

Note The user interface displays the values of the *number* and *extension* properties for the call forwarding entry when there isn't a description.

extension

String, which can contain up to 32 characters. The extension number of the call forwarding entry for the ad campaign.

The extension number can be preceded by one or more comma (,) characters, each of which represents a half-second pause, to allow enough time for the receiving phone to recognize the extension. If a value is not specified, an empty string is returned.

hrsmode

String. Indicates when calls are forwarded by the call forwarding entry for the ad campaign. The following table lists the values allowed for this property:

Value	Description
busnhrs	Forward calls only during business hours
afterhrs	Forward calls only after business hours
both	Default. Always forward calls, regardless of business hours

If a value is not specified, the default value is **both**.

number

String, which can contain up to 32 characters. The phone number of the call forwarding entry for the ad campaign.

status

String. Indicates the status of call forwarding entry for the ad campaign. The following table lists the values allowed for this property:

Value	Description
active	The call forwarding entry is active, and included when processing call forwarding entries.
inactive	The call forwarding entry is inactive, and excluded when processing call forwarding entries.

If a value is not specified, the default value is **active**.

step

Integer, which can contain a value between 1 and 20. The call forwarding step of the call forwarding entry for the ad campaign. Call forwarding entries are processed in sequential order, in which all call forwarding entries at a given call forwarding step are evaluated before moving to the next call forwarding step. If a value is not specified, the default value is 1.

Permissions

The API user must be a Group Admin of the group that the specified campaign belongs to. Note that users with the Account Admin permission are Group Admins of all groups in an account, and parent account users with the Client Admin permission are Group Admins of all groups in all sub-accounts.

Remarks

This method does not return forwarding timeouts, routing weights, or advanced routing rules for call forwarding entries. To retrieve that information for a specific call forwarding entry, use the ["ad.forw.get" on page 192](#) method.

Example

The following JSON-RPC request and response describe a successful call to the **ad.forw.list** method. The response returns two call forwarding entries for the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.list",
  "params": [
    "CmymM1NZcDoqwwA="
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "hrsmode": "busnhrs",
      "extension": "0005",
      "step": 1,
      "number": "2062370002",
      "status": "active",
      "assignto": "CmymD1NNGSxFmwD="
    }
  ]
}
```

```

        "descr": "Test description"
    },
    {
        "hrsmode": "afterhrs",
        "extension": "0010",
        "step": 2,
        "number": "2062370003",
        "status": "active",
        "assignto": "CmymDlNngSxFmwD-",
        "descr": "Test description"
    }
]
}

```

ad.forw.matchrule.set

Sets an expression representing the advanced routing rules for the specified call forwarding entry and ad campaign.

Request syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.matchrule.set",
  "params": [
    "cmpid",
    "number",
    "matchrule"
  ]
}

```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

number

Required string, which can contain exactly 10 characters. The telephone number of the specified call forwarding entry.

matchrule

Required string. An expression representing the advanced routing rules for the specified call forwarding entry and ad campaign. If a value is not specified, the default value is "".

Response syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,

```

```
"result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Remarks

Advanced routing rules provide more flexibility when designing a call forwarding strategy. For more information about advanced routing rules, see the "Advanced Routing Rules" of the *Call Tracking Ad Campaign* chapter.

Example

The following JSON-RPC request and response describe a successful call to the **ad.forw.matchrule.set** method. The request sets an advanced routing rule that matches all calls from Pennsylvania except those from LATA region 228 for the specified telephone number and ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.matchrule.set",
  "params": [
    "CA6phEtWMT47mgNN",
    "2064125430",
    "PA & !L228"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 2
}
```

ad.forw.new

Creates a new call forwarding entry that forwards inbound calls to the specified telephone number, for the specified ad campaign.

Request syntax


```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.new",
  "params": [
    "cmpid",
    "number",
    "descr"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

number

Required string, which can contain between 1 and 32 characters. The telephone number for the new call forwarding entry.

descr

Required string, which can contain up to 64 characters. The description of the new call forwarding entry.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Remarks

This method creates a new call forwarding entry, specifying default values for other properties, such as extension and step, for the new entry. The following table lists those properties and the default values used by this method. To change the values for any of these properties, use the appropriate method listed in the table to update the value for the new call forwarding entry.

Property	Default Value	Method
assignto	Empty string ("")	"ad.forw.assignto.set" on page 187
extension	Empty string ("")	"ad.forw.exten.set" on page 190

Property	Default Value	Method
hrsmode	both	"ad.forw.hrsmode.set" on page 194
matchrule	Empty string ("")	"ad.forw.matchrule.set" on page 199
status	active	"ad.forw.status.set" below
step	1	"ad.forw.step.set" on page 204
timeout	0	"ad.forw.timeout.set" on page 205
weight	0	"ad.forw.weight.set" on page 207

Example

The following JSON-RPC request and response describe a successful call to the **ad.forw.new** method. The request creates a new call forwarding entry with the specified description for the specified telephone number and ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.new",
  "params": [
    "CA6phEtWMT47mgNN",
    "2064125430",
    "Test call forwarding entry"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.forw.status.set

Sets the status of the specified call forwarding entry for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.status.set",
  "params": [
    "cmpid",

```

```

        "number",
        "status"
    ]
}

```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

number

Required string, which can contain exactly 10 characters. The telephone number of the specified call forwarding entry.

status

Required string, which can contain up to 16 characters. Indicates the status of a call forwarding entry for the ad campaign. The following table lists the values allowed for this property:

Value	Description
active	The call forwarding entry is active, and included when processing call forwarding entries.
inactive	The call forwarding entry is inactive, and excluded when processing call forwarding entries.

If a value is not specified, the default value is "active".

Response syntax

```

{
    "jsonrpc": "2.0",
    "id": 1,
    "result": result
}

```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.fow.status.set** method. The request sets the status of the call forwarding entry for the specified telephone number and ad campaign to "inactive", which excludes the call forwarding entry from evaluation when forwarding an incoming call.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.status.set",
  "params": [
    "CA6phEtWMT47mgNN",
    "2064125430",
    "inactive"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.forw.step.set

Sets the call forwarding step of the specified call forwarding entry, for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.step.set",
  "params": [
    "cmpid",
    "number",
    step
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

number

Required string, which can contain exactly 10 characters. The telephone number of the specified call forwarding entry.

step

Required integer, which can contain a value between 1 and 20. The call forwarding step of the specified call forwarding entry for the specified ad campaign. Call forwarding entries are processed in sequential order, in which

all call forwarding entries at a given call forwarding step are evaluated before moving to the next call forwarding step. If a value is not specified, the default value is 1.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.forw.step.set** method. The request assigns the call forwarding entry for the specified telephone number to call forwarding step 1 for the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.step.set",
  "params": [
    "CA6phEtWMT47mgNN",
    "2064125430",
    1
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.forw.timeout.set

Sets the time, in seconds, before the call forwarding time limit expires and ringing passes to the next call forwarding step (if any), or to voicemail (if enabled), of the specified call forwarding entry for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.timeout.set",
  "params": [
    "cmpid",
    "number",
    timeout
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

number

Required string, which can contain exactly 10 characters. The telephone number of the specified call forwarding entry.

timeout

Required integer, which can contain a value equal to or greater than 0. The time, in seconds, before the call forwarding time limit expires and ringing passes to the next call forwarding step (if any), or to voicemail (if enabled), for the call forwarding entry of the ad campaign. If a value is not specified, the default value is 0.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.forw.timeout.set** method. The request sets the timeout of the call forwarding entry for the specified telephone number to 24 seconds for the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.timeout.set",
  "params": [
    "CA6phEtWMT47mgNN",
    "2064125430",
    24
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.forw.weight.set

Sets the weighting of the specified call forwarding entry for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.weight.set",
  "params": [
    "cmpid",
    "number",
    weight
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

number

Required string, which can contain exactly 10 characters. The telephone number of the specified call forwarding entry.

weight

Required integer. The weighting of the call forwarding entry for the ad campaign. If more than one call forwarding entry is assigned to the same call forwarding step, calls are forwarded in proportion to the relative weight of each call forwarding entry at that call forwarding step. If a value is not specified, the default value is 0.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.forw.weight.set** method. The request sets the weight of the call forwarding entry for the specified telephone number to 50 for the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forw.weight.set",
  "params": [
    "CA6phEtWMT47mgNN",
    "2064125430",
    50
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result":
}
```

ad.forwtout.get

Gets the global call forwarding timeout for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
```



```
{
  "id": 1,
  "method": "ad.forwtout.get",
  "params": [
    "cmpid"
  ]
}
```

Request parameters

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. The global timeout, in seconds, before a call should be forwarded according to the call forwarding settings for the ad campaign.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Remarks

The global forwarding timeout limits the number of seconds a call can go unanswered. If the global forwarding timeout elapses, the caller is passed to voicemail, or the call is ended.

Example

The following JSON-RPC request and response describe a successful call to the **ad.forwtout.get** method. The response returns, in seconds, the global forwarding timeout of the ad campaign specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forwtout.get",
  "params": [
    "CA6phEtWMT47mgNN"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
```

```
}
  "result": 36
}
```

ad.forwtout.set

Sets the global call forwarding timeout for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forwtout.set",
  "params": [
    "cmpid",
    timeout
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

timeout

Required integer. The global timeout, in seconds, before a call should be forwarded according to the call forwarding settings for the ad campaign. If a value is not specified, the default value is 36.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Remarks

The global forwarding timeout limits the number a seconds a call can go unanswered. If the global forwarding timeout elapses, the caller is passed to voicemail, or the call is ended.

Example

The following JSON-RPC request and response describe a successful call to the **ad.forwtout.set** method. The request sets the global forwarding timeout to 36 seconds for the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.forwtout.set",
  "params": [
    "CA6phEtWMT47mgNN",
    36
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

Call Recordings, Settings, and Search

The methods in this section deal with call recordings, call-related settings on ad campaigns, and searching for calls in the call log.

Method	Description
<u>ad.recordcall.get</u>	Gets whether call recording is enabled for the specified ad campaign.
<u>ad.recordcall.set</u>	Sets whether call recording is enabled for the specified ad campaign.
<u>ad.usevmmail.get</u>	Gets whether a voicemail prompt is enabled for inbound calls to the specified ad campaign.
<u>ad.usevmmail.set</u>	Sets whether a voicemail prompt is enabled for inbound calls to the specified ad campaign.
<u>call.audio</u>	Gets a Base64-encoded string that contains the audio data of the specified call, in the specified format.
<u>call.audio.url</u>	Gets the URLs to the audio data for each call in the specified set of calls.
<u>call.custom.define</u>	Defines a call log custom field for the specified account.
<u>call.custom.set</u>	Sets the value of the specified call log custom field for the specified call log entry and account.

Method	Description
<u>call.get</u>	Gets the call log entry for the specified call.
<u>call.set</u>	Sets the user-modifiable details of the specified call.
<u>call.search</u>	Search all calls logged for the specified account, optionally constrained by a specified search filter.
<u>call.transcript.get</u>	Gets the text transcription for the specified call.

[ad.recordcall.get](#)

Gets whether call recording is enabled for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.recordcall.get",
  "params": [
    "campaign_id"
  ]
}
```

Request parameters

campaign_id

Required string. The unique, system-generated ID of the campaign.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": boolean
}
```

Response parameters

result

true if call recording is enabled for the specified ad campaign; otherwise, **false**.

Permissions

The API user must belong to the same group as the specified ad campaign. Note that users with the Account Admin permission are members of all groups in an account, and users with the Client Admin permission are members of all groups in all client accounts.

Example

The following JSON-RPC request and response show a successful call to the **ad.recordcall.get** method. The response indicates that call recording is disabled for the ad campaign specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.recordcall.get",
  "params": [
    "QrOXy0RPy_gATgEB"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": false
}
```

ad.recordcall.set

Enables or disables call recording for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.recordcall.set",
  "params": [
    "campaign_id",
    record_call
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

campaign_id

Required string. The unique, system-generated campaign ID of the specified ad campaign.

record_call

Required boolean. Specify **true** to enable call recording for the specified ad campaign. Specify **false** to disable call recording.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
```

```
    "result": integer
  }
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must be a Group Admin of the group that the specified campaign belongs to. Note that users with the Account Admin permission are Group Admins of all groups in an account, and parent account users with the Client Admin permission are Group Admins of all groups in all sub-accounts.

Example

The following JSON-RPC request and response shows a successful call to the **ad.recordcall.set** method. The request enables call recording for the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.recordcall.set",
  "params": [
    "QrOXy0RPy_gATgEB",
    true
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

ad.usevmail.get

Gets whether a voicemail prompt is enabled for inbound calls to the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.usevmail.get",
  "params": [
```

```
}
  "cmpid"
]
```

Request parameters

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required boolean. **true** if a voicemail prompt is enabled for inbound calls; otherwise, **false**.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.usevmail.get** method. The response indicates that the ad campaign specified in the request uses voicemail.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.usevmail.get",
  "params": [
    "QrOXy0RPy_gATgEB"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": true
}
```

ad.usevmail.set

Sets whether a voicemail prompt is enabled for inbound calls to the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.usevmail.set",
  "params": [
    "cmpid",
    usevmail
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

usevmail

Required boolean. **true** if a voicemail prompt is enabled for inbound calls; otherwise, **false**.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.usevmail.set** method. The request enables voicemail for the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.usevmail.set",
  "params": [
    "QrOXy0RPy_gATgEB",
    true
  ]
}
```



```
}
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

call.audio

Gets a Base64-encoded string that contains the audio data of the specified call, in the specified format.

If you (or someone in your company) will be listening to the recordings within 30 days, and you do not need to store all recordings, you may prefer to use ["call.audio.url" on page 219](#). The response time of **call.audio.url** is significantly faster than **call.audio** because instead of sending large audio data in the response, **call.audio.url** sends an array of URLs in the response that you can use to listen to the recordings. The player that is displayed in your browser when you access a recording via the URL allows you to download the recording if you need to save it.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "call.audio",
  "params": [
    "call_id",
    "audio_format"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

call_id

Required string. The unique, system-generated call ID. You can use the ["Response Syntax: Basic Response" on page 253](#) method to get the *call_id*.

audio_format

Required string. The format of the audio data for the specified call. The following table describes the allowed values:

Value	Description
mp3	Moving Picture Experts Group Layer-3 Audio (.mp3) format
wav	Windows Audio Video (.wav) format

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "Base64"
}
```

Response parameters

result

A Base64-encoded string that contains the audio data of the specified call in the specified format.

Permissions

The API user must belong to the same group as the ad campaign associated with the specified call. Note that users with the Account Admin permission are members of all groups in an account, and users with the Client Admin permission are members of all groups in all sub-accounts.

Remarks

This method does not set the Listened To flag in the call log.

If the specified call was not recorded, an error occurs. Use the ["ad.recordcall.get" on page 212](#) method to confirm that call recording is enabled for the ad campaign associated with the specified call, or use the ["call.get" on page 227](#) method to confirm that a call recording is available for the specified call.

Example

The following example shows a successful call to the **call.audio** method. Note that the response has been truncated for readability.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "call.audio",
  "params": [
    "CmymKlN65ctg4ACm",
    "mp3"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "/+AQZAAAY/VMgBqY8AJO..."
}
```

See also

» ["Call Recordings, Settings, and Search" on page 211](#)

call.audio.url

Gets URLs that link to recorded call audio for the specified array of calls. The URLs and tokens returned from this method are valid for 30 days. The player that is displayed in your browser when you access a recording via the URL allows you to download the recording if you need to save it.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "call.audio.url",
  "params": [
    ["call_id1", "call_id2", ..., "call_id200"],
    "audio_format"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

call_id array

Required array of strings. Each item in the array is the unique, system-generated call ID. You can use ["Response Syntax: Basic Response" on page 253](#) to get the call IDs

Note Do not include more than 200 call IDs in this array.

audio_format

Required string. The format of the audio data for the calls. The following table describes the allowed values:

Value	Description
wav	Windows Audio Video (.wav) format
mp3	Moving Picture Experts Group Layer-3 Audio (.mp3) format

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "call_id": "string",
      "token": "string",
      "url": "string"
    },
    {
      ...
    }
  ]
}
```

Response parameters

The result is an array that contains 0 or more objects, each of which contains the following key-value pairs:

call_id

String. The call ID.

token

Present only when the call has a recording. String. A unique, system-generated ID that identifies the call recording. The specified format for the audio data is appended to the token. Each request to **call.audio.url** with the same *call_id* will result in a new *token*.

url

Present only when the call has a recording. String. The URL (including the token) to the audio data for a call recording.

Permissions

The API user must belong to the same group(s) as the ad campaign(s) associated with the specified call(s). Note that users with the Account Admin permission are members of all groups in an account, and users with the Client Admin permission are members of all groups in all client accounts.

Remarks

- » This method does not mark any calls as Listened To in the call log, but calls are marked later when the URL-token values are used to fetch call audio.
- » If a call does not have a recording, you will see the following in the results for the call:
"error": "Call does not have a recording."
See the [Response](#) in the Example below.

Example

The following example returns the URLs that point to recordings for the calls specified in the request. Note that the third call does not have a recording.

Request

```
{
  "jsonrpc": "2.0",
  "method": "call.audio.url",
  "id": 1,
```

```

    "params": [
      [
        "CtjSIlMnkBY0KAAH",
        "CtjSMlMRGQYo9QAs",
        "Ch6WHVbokURjotLB"
      ],
      "wav"
    ]
  }

```

Response

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "url": "http://www.marchex.io/calls/audio?token=e20f4892c541de26e2c91d318baf5304.wav",
      "token": "e20f4892c541de26e2c91d318baf5304.wav",
      "call_id": "CtjSIlMnkBY0KAAH"
    },
    {
      "url": "http://www.marchex.io/calls/audio?token=9d7a5158c7d9f0c2406885f33befdc6d6.wav",
      "token": "9d7a5158c7d9f0c2406885f33befdc6d6.wav",
      "call_id": "CtjSMlMRGQYo9QAs"
    },
    {
      "call_id": "Ch6WHVbokURjotLB",
      "error": "Call does not have a recording."
    }
  ]
}

```

call.custom.define

Defines a call log custom field for the specified account. You use this method to define the name and data type of the custom field, and you use ["call.custom.set"](#) on [page 225](#) to specify the value for the custom field.

Request syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "call.custom.define",
  "params": [
    "account_id",
    field_number,
  ]
}

```

```

        "field_name",
        "field_type",
        [
            "menu_value_1",
            "menu_value_2",
            ...
            "menu_value_n"
        ]
    ]
}

```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

account_id

Required string. The unique, system-generated account ID of the account that you want to configure.

field_number

Required integer, which can contain a value between 1 and 20. The index number of the custom field.

field_name

Required string, which can contain up to 32 characters. The name of the custom field. The name is used to identify the custom field. The name is displayed in the user interface, and you specify the name when programmatically setting the value of the field with the **call.custom.set** method.

Note The custom field name must be unique across the entire account.

field_type

Required string. The type of the custom field. The custom field type restricts what values are allowed to be set for the custom field in the user interface and via **call.custom.set**. The following table describes the supported data types:

Value	Description
bool	A Boolean value. A drop-down list containing "yes" and "no" is displayed in the user interface, from which a user can select a value.
date	A date value. A date picker is displayed in the user interface, from which a user can select a date value.
float	A floating-point decimal numeric value. A text box is displayed in the user interface, in which a user can provide a floating-point decimal value.
int	An integer numeric value. A text box is displayed in the user interface, in which a user can provide an integer numeric value.
menu	An array of strings that you specify in the <i>menu items array</i> . A drop-down list containing the strings specified in the array is displayed in the user interface, from which a user can select a value.
text	A free-form text value. A text box is displayed in the user interface, in which a user can provide a free-form text value.

menu items array

Required array. If *field_type* is anything other than **menu**, specify an empty array. For the **menu** *field_type*, specify an array of strings. Each string in the array is displayed in the user interface as an item in the menu's drop-down list. If you set a value for the menu via **call.custom.set**, the string you specify in **call.custom.set** must exactly match (including case) one of the strings that you have specified in this array.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

Accounts can define up to 20 call log custom fields to support business-specific information for call log entries associated with that account.

Inheriting custom fields

By default, a client account inherits the call log custom fields from its parent account, and you see the inherited fields in the user interface in the client account's Call Log. If you use **call.custom.define** or the user interface to define a call log custom field for a client account, the client account no longer inherits call log custom fields from the parent account, and only the custom field defined in the client account will be displayed in the client account's Call Log.

Examples

The following JSON-RPC requests and responses describe successful calls to the **call.custom.define** method.

Example 1: Date custom field

The following request sets the name of call log custom field 1 for the specified account to **Review Date**, and configures the custom field to accept a date value. The new name of the custom field is displayed in the user interface, which now displays a date picker from which a user can choose a date value.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "call.custom.define",
  "params": [
    "QrOXxEE9-fATtgAD",
    1,
    "Review Date",
  ]
}
```

```
        "date",
        []
    ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

Example 2: Menu custom field

The following request sets the name of call log custom field 2 for the specified account to **Call Region**, and configures the custom field to accept a value from an array of acceptable values. The new name of the custom field is displayed in the user interface, which now displays a dropdown list of the acceptable values from which the user can choose.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "call.custom.define",
  "params": [
    "QrOXxEE9-fATtgAD",
    2,
    "Call Region",
    "menu",
    [
      "Central",
      "East North Central",
      "Northeast",
      "Northwest",
      "South",
      "Southeast",
      "Southwest",
      "West",
      "West North Central"
    ]
  ]
}
```

Response


```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

call.custom.set

Sets the value of the specified call log custom field for the specified call log entry and account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "call.custom.set",
  "params": [
    "account_id",
    "call_id",
    "field_name",
    "field_value"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

account_id

Required string. The unique, system-generated account ID containing the call log entry that you want to configure.

call_id

Required string. The unique, system-generated call ID of the call log entry to configure.

field_name

Required string. The name of the custom field to configure.

field_value

Required. The value of the campaign custom field. The value must be appropriate to the field type of the specified custom field, as described in the following table:

Field Type	Values
bool	A string with one of the following values: <ul style="list-style-type: none">» "yes" or "1"» "no" or "0"

Field Type	Values
date	<p>A date string in the format "DD Mon YYYY" where:</p> <ul style="list-style-type: none"> » DD is a two-digit day. » Mon is the first three letters of the month. (Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec) » YYYY is the four-digit year <p>For example: "24 Mar 2016"</p>
float	<p>A floating-point decimal numeric value, for example: 3.14159</p> <p>Note you may also specify the value as a string, for example: "5.7"</p>
int	<p>An integer numeric value. For example: 155</p> <p>Negative integers are allowed. For example: -33</p> <p>Note you may also specify the value as a string, for example: "161"</p>
menu	<p>A string that matches exactly (including case) one of the strings defined for the menu.</p> <p>If you used "call.custom.define" on page 221 to define the custom field, the value must be one of the strings that was specified in the <i>menu items array</i> request parameter.</p>
text	A string with a maximum length of 64 characters.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

You must use the name of the call log custom field to set its value. You cannot use the ordinal position of the custom field.

Example

The following JSON-RPC request and response describe a successful call to the **call.custom.set** method. The request sets the value of the call log custom field named **Call Region** to "Northwest" for the specified account and call log entry.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "call.custom.set",
  "params": [
    "QrOXxEE9-fATtgAD",
    "QrOX3UP6BJ5LZGNv",
    "Call Region",
    "Northwest"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

call.get

Gets the call log entry for the specified call.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "call.get",
  "params": [
    "call_id",
    {
      "extended": boolean,
      "include_dna": boolean,
      "include_dni_vars": boolean,
      "include_high_intent": boolean,
      "include_department": boolean,
      "include_sentiment": boolean,
      "include_spotted_keywords": boolean
    }
  ]
}
```

Request parameters

`call_id`

Required string. The unique, system-generated ID for the specified call.

`include_dna`

Optional: Boolean. A flag that indicates if you want the method to return DNA information: agent_ring_count, agent_speech_duration, caller_speech_duration and dna_class.

include_dni_vars

Optional: Boolean. A flag that indicates if you want the method to return specific variables collected from landing page URL or page variables.

include_high_intent

Optional: Boolean. A flag that indicates if you want to return high intent information about the call based on criteria you set.

include_department

Optional: Boolean. A flag that indicates if you want the method to return the department information for the call using the Department object.

include_sentiment

Optional: Boolean. A flag that indicates if you want the method to return sentiment score information about the positive or negative emotion for each speaker (agent and caller) on the call: agent_sentiment, caller_sentiment.

include_spotted_keywords

Optional: Boolean. A flag that indicates if you want the method to return a list of keywords - if any - for each speaker (agent and caller) that were found in the transcription of the call: agent_spotted_keywords, caller_spotted_keywords

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "dni_vars": {
      "var1": "string",
      "var2": "string"
    },
    "a_name": "string",
    "acct": "string",
    "address": {
      "address": "string",
      "city": "string",
      "name": "string",
      "state": "string",
      "zipcode": "string",
    },
    "agent_ring_count": number,
    "agent_sentiment": "string",
    "agent_speech_duration": "string",
    "agent_spotted_keywords": [string1,string2,...],
    "assigned_to": "string",
    "ass_user": "string",
    "answer_offset": number,
    "billable_minutes": number,
    "billing_acct": "string",
```

```

"c_name": "string",
"c2cid": "string",
"call_duration": number,
"call_end": "string",
"call_id": "string",
"call_s": "string",
"call_start": "string",
"call_status": "string",
"caller_name": "string",
"caller_number": "string",
"caller_sentiment": number,
"caller_speech_duration": "string",
"caller_spotted_keywords": [string1,string2,...],
"custom_id": "string",
"cmpid": "string",
"customer_type": "string",
"custom": [
    "string",
    "string",
    "string",
    "string",
    "string",
    "string",
    "string",
    "string",
    "string",
    "string",
    "string",
    "string",
    "string",
    "string",
    "string",
    "string",
    "string",
    "string",
    "string",
    "string",
    "string"
],
"camp_kw": "string",
"department": {
    "Unknown": boolean,
    "Service": boolean,
    "Sales": boolean
},
"disposition": "string",
"dna_class": "string",
"duration": "string",
"grpid": "string",
"high_intent": "string",
"forwardno": "string",

```

```

    "g_name": "string",
    "inbound_ext": "string",
    "inboundno": "string",
    "keyword": "string",
    "phone_type": "string",
    "listenedto": number,
    "note": "string",
    "rating": "string",
    "recorded": boolean,
    "revenue": "string",
    "numlookup_p": number,
    "redacted": boolean,
    "reverse_lookup_email": "string",
    "ring_duration": number
    "status": "string"
  }
}

```

Response parameters

Note Extended response parameters include additional data and are returned when either the extended or include_dna, include_dni_vars, include_high_intent, include_department, include_sentiment or include_spotted_keywords request parameters are set to true.

a_name

Extended. String. The name of the account associated with the call.

acct

String. The unique, system-generated ID of the account associated with the call.

address

Note The address object is in the results only when Reverse Address Lookup is enabled on the account containing the specified call.

String. The address object contains the following key-value pairs that are populated with information that is obtained by using the caller ID.

Important Additional information about the caller may not be available in all instances.

Key	String Values
<i>name</i>	The caller's name.
<i>address</i>	The street address.

Key	String Values
<i>city</i>	The city.
<i>state</i>	The state code.
<i>zipcode</i>	The postal code.

agent_ring_count

Extended. Number. The number of rings before the call was picked up by the agent, voice mail, or other system.

Returned when include_dna is set to true.

agent_sentiment

Extended. String. The sentiment score of the call based on the positive or negative emotion of the agent.

Returned when include_sentiment is set to true.

agent_spotted_keyowrds

String. A list containing the agent keywords spotted in the transcription of the call. If no agent keywords are found, an empty list is returned (0 items).

Notes

- » The total number of keywords that can be returned is 300 per speaker.
- » Each keyword is a string that can be a single word or a phrase that contains at most three words. The words are separated by a space.

Returned when include_spotted_keywords is set to true.

agent_speech_duration

Extended. String. The total amount of time in seconds that the agent spoke during the call. The returned value is a string containing a double-precision floating point number.

Returned when include_dna is set to true.

answer_offset

Number. The time in seconds between the time when the call connects to the Marchex network and when Marchex receives an answer signal. Includes ring and playfile time.

assigned_to

String. The unique, system-generated user ID of the user assigned to the call.

ass_user

Extended. String. The unique, system-generated user ID of the user assigned to the call.

billable_minutes

Extended. Number. The billable minutes for the call.

billing_acct

String. The unique, system-generated account ID of the billing account associated with the call.

c_name

Extended. String. The name of the ad campaign associated with the call.

call_duration

Number. The length of the call, in seconds.

.call_end

String. An ISO-8601 datetime value that represents the end date and time of the call. The time is returned in UTC, for example: "2015-03-23T08:05:25Z"

call_id

String. The unique, system-generated call ID of the call

call_s

Extended. String. An ISO-8601 datetime value that represents the start date and time of the call in the time zone that is set in the associated ad campaign. The datetime string is in the format:

"yyyy-mm-ddhh:mm:ss".

For example: "2015-03-16 20:28:32".

caller_sentiment

Extended. Number. The sentiment score of the call based on the positive or negative emotion of the caller.

Returned when `include_sentiment` is set to true.

call_start

String. An ISO-8601 datetime value that represents the start date and time of the call.

call_status

String. The status of the call. The following table lists the possible values:

Call Status	Definition
ANSWER	The call was answered at the Forward To phone number. Note that the call could have been answered by a person, an IVR , or sent to voice mail.
BUSY	The Forward To phone number was busy.
CANCEL/HANGUP	Indicates that the call ended before the Forward To phone number rang. When Extensions are enabled on an Ad Campaign, Hangup is logged when the caller hangs up in the Call Analytics IVR.
CONGESTION	Indicates a call not connecting because of a carrier issue, network congestion, or the Forward To number is out of service.
NOANSWER	The call was forwarded successfully to the Forward To number, but the call was not answered because the timeout value configured for the ad campaign elapsed, or the caller hung up.
NOFORWARDS	<p>The caller was disconnected before Call Analytics could dial out to the Forward To number. This can happen when:</p> <ul style="list-style-type: none">» The caller did not match any forwarding rules.» No forwarding rules have been configured on the Ad Campaign. <p>When Extensions are enabled on an Ad Campaign, NOFORWARDS is logged when Call Analytics ends the call because of a timeout. This can happen when:</p> <ul style="list-style-type: none">» The caller never selected an option in the Call Analytics IVR.» The caller selected an invalid option in the IVR.
VOICEMAIL	The call was transferred to the Call Analytics Voicemail system.
WINDDOWN	The call reached the Call Analytics ad campaign during the wind down period.

caller_name

String. The caller's name, if available, obtained from the caller ID information on the call.

caller_number

String. The caller's phone number, if available, obtained from the caller ID information on the call.

caller_sentiment

Extended. Number. The sentiment score of the call based on the positive or negative emotion of the caller.

caller_speech_duration

Extended. String. The total amount of time in seconds that the caller spoke during the call. The returned value is a string containing a double-precision floating point number.

Returned when `include_dna` is set to true.

caller_spotted_keywords

Extended. String. A list containing the caller keywords spotted in the transcription of the call. If no caller keywords are found, the list returns (0 items).

Notes

- » The total number of keywords that can be returned is 300 per speaker.
- » Each keyword is a string that can be a single word or a phrase that contains at most three words. The words are separated by a space.

Returned when `include_spotted_keywords` is set to true.

cmpid

String. The unique, system-generated ID for the ad campaign associated with the call.

customer_type

String. Identifies the caller as commercial, residential, or other.

Note `Customer_type` is only available when reverse lookup is enabled on the account.

Note A blank or null value may be returned if data is not available.

camp_kw

Extended. String. When the keyword response parameter associated with the call is empty, `camp_kw` contains the name of the ad campaign. When the keyword contains a value, `camp_kw` contains the campaign name and the keyword separated by a colon (:) and a space. For example: "Company National Campaign: widgets"

custom

An array of 20 strings that contain the values of the Call Log Custom Fields set on the specified call. The Call Log Custom Field names and possible values are defined on the account. You can set the value for the Call Log Custom Fields on a call either through the Call Analytics user interface or use the `call.set` method. The position of each string in the array corresponds to the "Custom n" field (where n is between 1 and 20) as it appears on the account's Call Log Custom Field page in the Call Analytics user interface. Empty strings ("") are returned for the custom fields that have not had a value set on the call, or that have not been configured on the account.

Call Log Custom Fields

Custom 1:	qualified lead	<div>yes ▼ yes no</div>	Edit	Remove
Custom 2:	Scheduled		Edit	Remove
Custom 3:	Wrong Number	<div>yes ▼</div>	Edit	Remove
Custom 4:	Test by Amber	<div>yes ▼</div>	Edit	Remove
Custom 5:	Status	<div>Good ▼</div>	Edit	Remove
Custom 6:	Person	<div></div>	Edit	Remove
Custom 7:	Good?	<div>Good! ▼</div>	Edit	Remove
Custom 8:		<div>Yes ▼</div>	Edit	Remove
Custom 9:		not configured	Configure	
Custom 10:		not configured	Configure	
Custom 11:		not configured	Configure	

custom_id

Extended. String. The value of the Custom reference ID field on the Account Settings + Company Info and Setting page in the Call Analytics user interface.

department

Extended. Boolean. Identifies which department received the call depending on how this value is configured. Example departments: sales, service, support.

disposition

String. Maps to the Tag field in the call log in the Call Analytics user interface. The possible values are set on the ad campaign associated with the call. The value for the specified call is set either by someone reviewing the calls in the call log in the Call Analytics user interface or via the `call.set` method.

dna_class

Extended. String. Use of this filter in your request returns only calls with the provided Call DNA classification.

Returned when include_dna is set to true.

Note If you want Call DNA information returned for all calls, set include_dna to true and do not include dna_class in the request.

Note To restrict the search results to the Call DNA classification, use one of the following strings, as it appears (including case):

Classification	Definition
Conversation	There was a dialog between the caller and agent.
Voicemail	The caller reached a voice mail recording and left a message for the advertiser.
Wrong Number	The caller reached the wrong business and ended the phone call after a brief exchange.
Hold-Time Abandon	The caller was put on hold either by an automated phone tree (IVR) or an agent and hung up the phone before the agent came back on the line.
IVR/Voicemail Abandon	The caller reached a voice mail recording or automated phone tree (IVR) and hung up the phone without leaving a message or speaking to an agent.
Ring Transfer Abandon	The call was answered, then the caller was transferred either by an IVR or an agent, but the caller hung up the phone before speaking to an agent.
Unknown	The call pattern did not match any of the known Call DNA classifications.

dni_vars

Extended. String. The multi variable data associated with the call. This is applicable only for calls associated with dynamic-tracking ad campaigns that are configured for multi-variable collection.

duration

Extended. String. The duration of the call, represented as a string showing the hours, minutes, and seconds of the call duration in "hh:mm:ss" format.

forwardno

String. The phone number to which the call was forwarded.

g_name

Extended. String. The name of the ad group associated with the call.

grp_id

String. The unique, system-generated group ID of the ad group associated with the call.

high_intent

String. Indicates a call that shows high interest or high probability for conversion.

Returned when include_high_intent is set to true.

inbound_ext

String. The inbound extension number of the call. This is applicable only for calls associated with an ad campaign that has an IVR configured. The inbound_ext value is the number the caller entered in the IVR.

inbound_no

String. The inbound phone number, also referred to as the call tracking number. The phone number that the caller dialed.

keyword

String. The keyword associated with the call. This is applicable only for calls associated with dynamic-tracking ad campaigns.

phone_type

String. Identifies incoming call as landline, mobile, or other.

Note Phone_type is only available when reverse lookup is enabled on the account.

Note A blank or null value may be returned if data is not available.

listenedto

Extended. Number. The value is set to "1" if a user has listened to the call; otherwise, the value is set to "0".

note

Extended. String. Optional information about the call. The value for the specified call is set either by someone reviewing the calls in the call log in the Call Analytics user interface or via the call.set method.

numlookup_p

Extended. Number. Returns a "1" when Reverse Call Lookup is enabled on the account and a "0" when Reverse Call Lookup is not enabled.

rating

String. The rating of the call. The rating value is set either by someone reviewing the calls in the call log in the Call Analytics user interface or via the call.set method.

recorded

Boolean: true if the call was recorded; otherwise, false.

redacted

Extended. Boolean. Indicates if redaction has occurred in the recording. For accounts with Call Recording Redaction enabled:

- » The value is True if there are > 0 seconds of redaction in the recording.
- » The value is False if nothing was redacted in the recording or if Call Recording Redaction is not enabled on the account.

revenue

String. The amount of revenue, as set by the user, generated by the call.

reverse_lookup_email

String. The callers email address.

Note Email address is available for accounts that have been configured to receive it. Even when enabled, email address information may not be available for all callers.

.ring_duration

Number. The time in seconds between the time when the call is connected to the destination network and when is answered by a person, IVR or voice mail system. Includes ring time only.

status

Extended. String. The status of the ad campaign where the value returned is either "active" or "disabled".

Permissions

The API user must belong to the same group as the ad campaign associated with the specified call. Note that users with the Account Admin permission are members of all groups in an account, and users with the Client Admin permission are members of all groups in all sub-accounts.

Example request

In the following example, the *address* object contains reverse lookup information because Reverse Address Lookup is enabled on the account. In the *custom* array, only the Call Log Custom Fields Custom 1, Custom 2, and Custom 5 were set on the specified call, and empty strings are returned for the other custom fields.

Example Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "call.get",
  "params": [
    "Qr0XzUKkohMw9wDK",
    {
      "extended": true,
      "include_dna": true,
      "include_dni_vars": true,
      "include_high_intent": true,
      "include_department": false,
      "include_sentiment": true,
      "include_spotted_keywords": true
    }
  ]
}
```

Example Response

```
{
  "result": {
    "redacted": false,
    "numlookup_p": 1,
    "inboundno": "8443316584",
    "cmpid": "fwABAVzGxTsHmAB7",
    "billable_minutes": 1,
    "call_end": "2019-09-17T05:59:03Z",
    "forwardno": "8886648912",
    "call_status": "ANSWER",
    "status": "active",
    "recorded": false,
    "call_duration": 29,
    "billing_acct": "fwABAVzGhstDawCf",
    "note": "Good call",
    "caller_name": "ESCONDIDO CA",
    "inbound_ext": "",
    "agent_speech_duration": 5.54,
    "caller_sentiment": "",
    "g_name": "default",
    "c2cid": "",
    "customer_type": "residential",
    "custom": [
      "",
      "",
      "",
      "",
      "",
      "",
      ""
    ]
  }
}
```

```

    "",
    "",
    "",
    "",
    "",
    "",
    "",
    "",
    "",
    "",
    "",
    "",
    "",
    "",
    "",
    ""
],
"revenue": "",
"address": {
  "city": "MOUNTAIN VIEW",
  "state": "CA",
  "zipcode": "94043",
  "address": "1600 AMPHI PKWY \n ",
  "name": "GUY GOGGLE"
},
"acct": "fwABAVzGwtXMTwAi",
"caller_number": "7607058888",
"call_id": "Cs2WE12AdglHNAp_",
"answer_offset": 4,
"camp_kw": "Child-Camp-1",
"agent_ring_count": 5,
"ring_duration": 2,
"disposition": "",
"agent_sentiment": 5.77855132706463,
"a_name": "A4T-325-Child-1",
"reverse_lookup_email": "",
"assigned_to": "",
"rating": "",
"dna_class": "Unknown",
"ass_user": "",
"agent_spotted_keywords": [],
"caller_spotted_keywords": [],
"c_name": "Child-Camp-1",
"custom_id": "durvesh.bhole@afourtech.com",
"listenedto": 0,
"phone_type": "mobile",
"grpid": "fwABAVzGwtlMTwAj",
"caller_speech_duration": 10.64,
"duration": "00:00:29",
"call_start": "2019-09-17T05:58:33Z",
"call_s": "2019-09-17 01:58:33",
"high_intent": "",

```



```
        "keyword": ""  
    },  
    "id": 1,  
    "jsonrpc": "2.0"  
}
```

[PDF DOWNLOADS](#)



[Call Analytics JSON-RPC API Guide](#)

call.set

Sets the user-modifiable details of the specified call.

Request syntax

Required parameters are in bold. The object is required even if it is empty.

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "call.set",
  "params": [
    "call_id",

    {
      "assigned_to": "string",
      "custom": [
        "string", "string", ...
      ],
      "custom1": "string",
      ...
      "custom20": "string"
      "disposition": "string",
      "note": "string",
      "rating": integer,
      "revenue": number
    }
  ]
}
```

Request parameters

call_id

Required string. The unique, system-generated ID of the call. Use the ["Response Syntax: Basic Response" on page 253](#) method to get call ID.

A required object that contains the following key-value pairs:

assigned_to

Optional string. The unique, system-generated ID of the user assigned to the call.

custom

Optional array of 1 to 20 strings. The values of the Call Log Custom Fields to set on the specified call. The Call Log Custom Field names and possible values are defined on the account. The position of each string in the array corresponds to the "Custom *n*" field (where *n* is between 1 and 20) as it appears on the account's Call Log Custom Field page in the Call Analytics user interface.

Call Log Custom Fields

Custom 1:	qualified lead	<div>yes ▼ yes no</div>	Edit	Remove
Custom 2:	Scheduled		Edit	Remove
Custom 3:	Wrong Number	<div>yes ▼</div>	Edit	Remove
Custom 4:	Test by Amber	<div>yes ▼</div>	Edit	Remove
Custom 5:	Status	<div>Good ▼</div>	Edit	Remove
Custom 6:	Person	<div></div>	Edit	Remove
Custom 7:	Good?	<div>Good! ▼</div>	Edit	Remove
Custom 8:		<div>Yes ▼</div>	Edit	Remove
Custom 9:		not configured	Configure	
Custom 10:		not configured	Configure	
Custom 11:		not configured	Configure	

custom1 - custom20

Optional string. The value of Call Log Custom Field *n* (where *n* is between 1 and 20) to set on the specified call. You can use either the key-value pairs or the *custom* array to set the Call Log Custom Fields on the specified call. If you include both the *custom* array and one (or more) of the *customn* key-value pairs in the request, the method uses the values in the *custom* array.

Note the following:

- » You do not need to set every custom field on a call.
- » To clear the value of a custom field on a call, enter an empty string.
- » The strings that you specify in the *custom* array or in the *customn* key-value pairs must have already been configured on the account. To obtain the values that you can set, go to the account's Call Log Custom Field page in the Call Analytics user interface.
- » If you pass in an invalid value for a custom field, the method returns success but does not set the value on the call.

disposition

Optional string. The tag of the call. The *disposition* maps to the Tag field in the call log in the Call Analytics user interface. To obtain the possible values that you can set open the ad campaign associated with the call in the Call Analytics user interface, and go to the Tag tab.

note

Optional string. Optional information about the call.

rating

Optional integer. The rating of the call. Valid values are the integers **1** through **10**.

revenue

Optional number. The amount of revenue generated by the call.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must belong to the same group as the ad campaign associated with the specified call. Note that users with the Account Admin permission are members of all groups in an account, and users with the Client Admin permission are members of all groups in all sub-accounts.

Example

The following example request reassigns the call to a different user and updates various fields, including the first custom field for that call log entry.

Request

```
{
  "jsonrpc": "2.0",
  "method": "call.set",
  "id": 1,
  "params": [
    "CmymKlON-awWBQBT",
    {
      "assigned_to": "CmymDlNngSxFmWD-",
      "disposition": "Test tag",
      "note": "Updated via API",
      "rating": "5",
      "revenue": "1.00",
      "custom": [
        "qualified"
      ]
    }
  ]
}
```

```
]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

call.search

Searches the call log of the specified account. You can refine your search using the search filters defined below.

Note If the account receives a large number of calls, we recommend that you limit the start and end time for the search to a few hours to avoid latency. Alternately, you can enable pagination for large result sets. For accounts that do not receive many calls, you can expand the date/time range. Each call in the call log has a unique `call_id` value. You can track the returned `call_ids` and remove duplicate calls if needed. To make sure that you have not missed any entries in the call log, you may want to set the start time slightly earlier than the end time of the last call log entry returned by `call.search`.

Request syntax

This includes parameters for **both** **basic** and **extended** response types.

Note Required parameters are in **bold**.

Request parameters: Data return

Use these parameters to determine what `call.search` returns for all calls meeting the filtering criteria sample structure.

```
"extended": Boolean,

"include_dna": Boolean,
"include_dni_vars" : Boolean,

"include_high_intent": Boolean,

"include_department": Boolean,

"include_sentiment": Boolean,

"include_spotted_keywords": Boolean
```

Request parameters: Targeted filtering

Use these elements to target or filter specific call elements.

```
"start": "string",
"end": "string",
"exact_times": Boolean,
"assto": "string",
"call_boundary": "string",
"callerid": "string",
"cmpid": "string",
"dispo": "string",
"dna_class": "string",
"grpId": "string",
"keyword": "string",
"min_duration_secs": integer,
"status": "string",
"subacct": Boolean,
"spotted_keywords": {
  "agent_spotted_keywords": ["keyword1", "keyword2", ],
  "caller_spotted_keywords": ["keyword1", "keyword2"]
}
```

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "call.search",
  "params": [
    "account_id",
```

```

{
  "start" : "string",
  "end" : "string",
  "extended": Boolean,
  "include_dni_vars" : Boolean,
  "include_dna": Boolean,
  "include_high intent": Boolean,
  "include_department": Boolean,
  "include_sentiment": Boolean,
  "include_spotted_keywords": Boolean,
  "call_status": string,
  "dna_class": "string",
  "nx": "boolean, 1 or 0",
  "limit": "integer",
  "spotted_keywords": {
    "agent": {
      {
        "keywords":["keyword1", "keyword2",...]
      },
      "caller":
      {
        ["keyword1", "keyword2", ... ]
      }
    },
    "subacct": true
  }
}
]
}

```

Request parameters

account_id

Required string. This is a unique, 16-digit, system generated, ID for the account you wish to retrieve and analyze calls for.

start, end

Recommended strings; The start and end date/time of the search range.

Note As a best practice, always specify the start and end date/time for the search.

Although the method may succeed if you do not provide the start and end times, if there are many calls in the call log, you run the risk of the method timing out by not specifying the start and end times for the search.

1. Enter a valid ISO 8601 datetime value. Include an offset from UTC to specify the time zone. If you do not include an offset from UTC in the string, the time is assumed to be in UTC.

For example, to get all the calls on February 28, 2016 in Pacific Standard Time:

```
start = "2016-02-28T00:00:00-08:00"
```

```
end = "2016-02-28T23:59:59-08:00"
```

Note During daylight savings, you need to change the UTC offset in both start and end. For example, 6:30 PM Pacific Standard Time is "T18:30:00-08:00" and 6:30 PM Pacific Daylight Time is "T18:30:00-07:00".

Note It also appears that the start and end values do not have to be included at the top of the method and can appear in any location within the request. If you have included one or more of the parameters that tell call.search to return an extended response (refer to Response for more information), you must include the exact_times parameter in the request and set it to true. Otherwise the datetime values in start and end are treated as dates and the time portion is ignored.

Although a basic response does not need the exact_times, we recommend that you do include it.

For example, to find calls on February 28, 2016 between 6:30 PM and 7:15 PM Pacific Standard Time and get an extended response, specify the following values:

```
start = "2016-02-28T18:30:00-08:00"
```

```
end = "2016-02-28T19:15:00-08:00"
```

```
exact_times = true
```

```
extended = true
```

Note Call Analytics stores the datetime value for calls in its underlying database in UTC, but in the call log displayed in the Call Analytics user interface and in reports, the time is converted to the time zone setting in the ad campaign associated with the call.

assto

Optional: string; The unique, system-generated ID for a user that is assigned to a call.

call_boundary

Optional: string; Determines the time range when filtering the call log, the call start time, and the call end time.

Call_boundary options

Value	Description
start	Default. Include the call in the results if the start date and time of the call falls within the dates and times specified in the start and end parameters.

Value	Description
end	Include the call in the results if the end date and time of the call falls within the dates and times specified in the start and end parameters.

Note If a parameter is not specified when a value is entered, the "start" parameter is the default value assumed.

groupid

Optional: String; The unique, 16 character system-generated ad group ID for the call.

callerid

Optional: string; By including this value in the request, the system will limit the calls returned to the user ID in the string.

cmpid

Optional string; The unique, system-generated, campaign ID.

dispo

Optional: string; The tag can be specified in an ad campaign that can be assigned to a call or calls in the call log. This allows you to segment the data in the reporting to study specific types of calls.

- » You can add a tag to an ad campaign by using the Call Analytics user interface or by calling the `ad.dispo.new` method.
- » You can also assign a tag to a call (or calls) by using the Call Analytics user interface or by calling the `call.set` method.

dna_class

Optional: Boolean; Use of this filter in your request returns only calls with the provided CallDNA classification.

Note This property is used only if the `include_dna` request parameter is set to true; otherwise, the value of this property is ignored.

If you want Call DNA information returned for all calls, set `include_dna` to true and do not include `dna_class` in the request.

To restrict the search results to the Call DNA classification, use one of the following strings, as it appears (including case):

Classification	Definition
Conversation	There was a dialog between the caller and agent.
Voicemail	The caller reached a voice mail recording and left a message for the advertiser.
Wrong Number	The caller reached the wrong business and ended the phone call after a brief exchange.
Hold-Time Abandon	The caller was put on hold either by an automated phone tree (IVR) or an agent and hung up the phone before the agent came back on the line.
IVR/Voicemail Abandon	The caller reached a voice mail recording or automated phone tree (IVR) and hung up the phone without leaving a message or speaking to an agent.
Ring Transfer Abandon	The call was answered, then the caller was transferred either by an IVR or an agent, but the caller hung up the phone before speaking to an agent.
Unknown	The call pattern did not match any of the known Call DNA classifications.

`exact_times`

Optional: Boolean; Indicates the start and end parameters.

To search for a particular time period (and assuming you have included one or more of the parameters that tell `call.search` to return an extended response), you must include the `exact_times` parameter in the request and set it to `true`.

If the datetime values in `start` and `end` are treated as dates then the time portion is ignored. Refer to Response for more information.

Note You do not need to include `exact_times` for a basic response, but it is available if you choose to use it.

`extended`

Optional: Boolean; Set `extended` to `true` to return the basic information and the additional (extended) information, about each call. You must also include the `exact_times` parameter in the request and set it to `true`. If the parameter is not included or is set to `false`, the response includes only basic information about each call. Refer to Basic Response Syntax for the list of basic call properties.

Refer to the Extended Response Syntax for a list of the additional information properties that can be returned.

`include_dna`

Optional: Boolean; Specify `true` to return the extended call properties and the Call DNA information. You must also include the `exact_times` parameter in the request and set it to `true`. If the parameter is not included or is set to `false`, Call DNA information is not returned in the results.

Refer to the Include Call DNA Response Syntax for a list of the additional available properties.

Note Do not include this parameter if Call DNA is not enabled on the specified account.

If this is set to `true` and the specified account does not have Call DNA enabled, an error occurs.

`include_dni_vars`

Optional: Boolean; Specify `true` to return specific variables collected from landing page URL or page variables. If the parameter is not included or is set to `false`, multiple variable information is not returned in the results.

`include_high_intent`

Optional: Boolean; Specify `true` to return high intent information about the call, in other words whether the call is likely to result in a conversion based on criteria you set. If the parameter is not included or is set to `false`, the high intent information is not returned in the reporting results.

`include_department`

Optional: Boolean; Specify `true` to return specific department information for each call. If the parameter is not included or is set to `false`, department information is not returned in the results.

`include_sentiment`

Optional: Boolean; Specify `true` to return sentiment score information, in other words information about the positive or negative emotion for each speaker (agent and caller) on the call. If the parameter is not included or is set to `false`, the call sentiment score is not returned in the results.

`include_spotted_keywords`

Optional: Boolean; Properties plus a list of keywords—if any—for each speaker (agent and caller) that were found in the transcription of the call.

Note You must also include the `exact_times` parameter in the request and set it to `true`.

If the parameter is not included or is set to false, the "spotted" keywords (that is, the keywords found in a call transcription) are not returned in the results.

Refer to Include Keyword Spotting Response Syntax for a list of the additional properties that are returned.

Note Do not include this parameter if Call Transcription is not enabled on the specified account. An error occurs if this parameter is set to true and the specified account does not have Call Transcription enabled.

keyword

Optional: string; The keyword or the Custom URL Parameter (if specified in the dynamic tracking rewrite) associated with the call. This is relevant only for calls associated with dynamic tracking ad campaigns.

Note This parameter is not related to call transcription keywords used in keyword spotting.

min_duration_secs

Optional: integer; The minimum number of seconds for the call duration. If specified, only calls greater than or equal to the specified value will be returned.

limit

Optional: integer; Overrides the default limit is 1,000 records returned when paging is enabled. Maximum value for the *limit* parameter is 20000.

nx

Optional: boolean; Enables paging. The default limit is 1,000 records returned when paging is enabled.

Example

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "call.search",
  "params": [
    "QrOxy0FO7lkf_AAB",
    {
      "nx": "1",
    }
  ]
}
```

Note This note applies to scenarios where pagination is not enabled with large data sets. Marchex recommends that you use pagination when 10,000 or more records are returned.

spotted_keywords

Optional: object; If specified, only calls containing one or more of the specified call transcription keywords are returned.

Note The spotted_keywords parameter is independent of include_spotted_keywords.

Note You can include the spotted_keywords parameter in the request to filter the results, but you do not need to specify the include_spotted_keywords parameter unless you want the additional information that is included in the response.

You can specify keywords for either the agent, the caller, or both. Call the acct.keyword_spotting.word_list.get method to get a list of the currently defined keywords for the account. The spotted_keywords object contains the following key-value pairs in the following table.

Field	Description
agent	Filter the calls based on keywords spoken by the agent. This is an optional object that contains an array of strings called keywords. The strings that you specify in the keywords array must have already been configured as agent keywords in the specified account.
caller	Filter the calls based on keywords spoken by the caller. This is an optional object that contains an array of strings called keywords. The strings that you specify in the keywords array must have already been configured as caller keywords in the specified account.

Note Keywords are not spotted retroactively; only calls that arrive after the keywords have been set will have keywords flagged in the transcriptions.

Call transcription must be enabled on the account to set keywords. If call transcription is not enabled on the account, an error is returned. Contact Client Services to get call transcription enabled.

status

Optional: string; Provides the status of the call. To restrict the search results to a call or calls, enter one of the following strings, exactly as given (including case), in the following table:

Call Status	Definition
ANSWER	The call was answered at the Forward To phone number. Note that the call could have been answered by a person, an IVR, or sent to voice mail.
BUSY	The Forward To phone number was busy.
CANCEL or HANGUP	Indicates that the call ended before the Forward To phone number rang. Cancel or Hangup is logged for the following reasons: The caller hung up in the Call Analytics IVR configured for the ad campaign. Either the caller never selected an option in the Call Analytics IVR or selected an invalid option, which resulted in the call ending because of a timeout. The caller was disconnected before Call Analytics could dial out to the Forward To number.
CONGESTION	Indicates a call not connecting because of a carrier issue, network congestion, or the Forward To number is out of service.

Call Status	Definition
NOANSWER	The call was forwarded successfully to the Forward To number, but the call was not answered because the timeout value configured for the ad campaign elapsed, or the caller hung up.
NOFORWARDS	The caller did not match any forwarding rules, or no forwarding rules have been configured on the ad campaign.
WINDDOWN	The call reached the Call Analytics ad campaign during the wind down period.

subacct

Optional: Boolean; Set as true to search all client account calls for the specified parent account. If not included or false, only calls for the specified account are searched.

Response syntax: Response

The call.search method provides different responses depending on the values specified for the following request parameters:

- » dna_class
- » exact times
- » extended
- » include_department
- » include_dna
- » include_dni_vars
- » include_high_intent
- » include_sentiment
- » include_spotted_keywords

Response Syntax: Basic Response

Response syntax

If the above parameters are not included in the request, or if they are included but set to false, then the method returns a basic response.

Note The basic response doesn't include detailed information, such as call custom fields, or optional information, such as Call DNA information. Refer to Basic Response Syntax for details.

The following call properties are returned when:

1. The extended or include_dna request parameters are not included in the request.
2. The parameter is set to false.

```
{
  "jsonrpc": "2.0",
  "id": 1,
```

```
"next_token": string,
"result": [
  {
    "acct": "string",
    "answer_offset": integer,
    "assigned_to": "string",
    "call_duration": integer,
    "call_end": "string",
    "call_id": "string",
    "call_start": "string",
    "call_status": "string",
    "caller_name": "string",
    "caller_number": "string",
    "cmpid": "string",
    "disposition": "string",
    "forwardno": "string",
    "grp_id": "string",
    "inbound_ext": "string",
    "inboundno": "string",
    "keyword": "string",
    "numlookup_p": Boolean: 1 or 0,
    "rating": "string",
    "recorded": Boolean,
    "revenue": "string",
    "ring_duration": integer
  }
]
}
```

Response parameters: Basic Response

acct

The unique, system-generated ID for the account associated with the call.

answer_offset

Number. The time in seconds between the time when the call connects to the Marchex network and when Marchex receives an answer signal from the destination network. Includes ring and playfile time.

assigned_to

The unique, system-generated user ID for the user assigned to the call.

call_duration

The duration of the call measured in seconds.

call_end

An ISO-8601 datetime value that represents the end date and time of the call. The time is returned in UTC, for example: "2015-03-23T08:05:25Z"

call_id

The unique, system-generated ID for the call.

call_start

An ISO-8601 datetime value that represents the start date and time of the call. The time is returned in UTC, for example: "2015-03-23T08:05:04Z"

call_status

The status of the call. Refer to the Call Status Definitions table for details.

caller_name

The caller's name. This is the name provided by the telephone network with the call.

caller_number

The caller's phone number. This is the caller ID provided by the telephone network with the call.

cmpid

The unique, system-generated campaign ID for the ad campaign associated with the call.

disposition

The tag or disposition of the call. Refer to the dispo request parameter contained in the Search Filters section of this document, for additional information about creating and setting tags.

forwardno

The phone number to which the call was forwarded.

grpid

The unique, system-generated group ID of the ad group associated with the call.

inbound_ext

The inbound extension number.

inboundno

The inbound phone number, also referred to as the call tracking number. The phone number that the caller dialed.

keyword

The keyword associated with the call (if keyword level tracking is enabled for the call's ad campaign).

next_token

Returned when the nx parameter for paging is enabled and all results have not been returned. The value returned can be used to request the next data set.

Example

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "call.search",
  "params": [
    "QrOxY0FO7lkf_AAB",
    {
      "nx": "1",
      "next_token": "u1GohCsP0Xvt5g1khc_/4aYEJoC6RmttYvTRZe/gtJM",
    }
  ]
}
```

Note next_token is not returned when there are no more results.

Note If an error occurs, an HTTP status code other than 200 is returned.

numlookup_p

Returns a "1" when Reverse Call Lookup is enabled on the account and a "0" when Reverse Call Lookup is not enabled.

rating

The rating of the call, which is an integer from 1 to 10. You can assign a rating to a call (or calls) by using the Call Analytics user interface or by calling the call.set method.

recorded

The value is set to true if the call was recorded; otherwise, it is set to false.

revenue

The amount of revenue, generated by the call. You can assign revenue to a call (or calls) by using the Call Analytics user interface or by calling the `call.set` method.

ring_duration

Number. The time in seconds between the time when the call is connected to the destination network and when is answered by a person, IVR or voice mail system. Includes ring time only.

Response Syntax: Extended Response

If the `extended_request` parameter is set to `true`, the method returns an extended response, which includes the basic response information and the extended response information about the calls, such as custom fields.

Include Call DNA Response

If the `include_dna` request parameter is set to `true` (and the Call DNA feature is enabled for the specified account), the resulting response includes the extended response and Call DNA information for the call log entries returned by the method. Refer to the [Include Call DNA Response Syntax](#) for details.

Include Call Transcription Keywords Response

If `include_spotted_keywords` parameter is set to `true` (and the Call Transcription feature is enabled for the specified account), the response includes all extended response properties and the keywords that were found in the call transcription (the "spotted" keywords). Refer to [Include Keyword Spotting Response Syntax](#) for details.

The extended response includes the basic call properties described above and the following additional properties:

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "a_name": "string",
      "ass_user": "string",
      "billable_minutes": integer,
      "c_name": "string",
      "call_s": "string",
      "camp_kw": "string",
      "custom_id": "string",
      "custom1": "string",
      "custom2": "string",
```

```
"custom3": "string",
"custom4": "string",
"custom5": "string",
"custom6": "string",
"custom7": "string",
"custom8": "string",
"custom9": "string",
"custom10": "string",
"custom11": "string",
"custom12": "string",
"custom13": "string",
"custom14": "string",
"custom15": "string",
"custom16": "string",
"custom17": "string",
"custom18": "string",
"custom19": "string",
"custom20": "string",
"customer_type": "string",
"dna_class": "string",
"duration": "string",
"g_name": "string",
"listenedto": Boolean: 0 or 1,
"note": "string",
"numlookup": "string",
"phone_type": "string",
"redacted": Boolean,
"status": "string"
"high_intent": "Boolean",
```

```

    "department": {
      "Service": "Boolean",
      "Sales": "Boolean"
    },
    "agent_sentiment": "integer"
    "caller_sentiment": "integer"
  }
]
}

```

Note This is the response if you want these values; add them to your JSON request and you'll get an extended response.

Response parameters

The Extended response parameters are returned when either the extended or include_dna request parameters are set to true.

The Extended response includes the Basic properties described above and the following additional properties:

a_name

String. The name of the account associated with the call.

ass_user

String. The name of the user assigned to the call.

billable_minutes

Number. The billable minutes for the call.

c_name

String. The name of the ad campaign associated with the call.

call_s

String. The start date and time of the call, in the time zone that is set in the associated ad campaign. The datetime string is in the format:

"yyyy-mm-ddhh:mm:ss".

For example: "2015-03-16 20:28:32".

camp_kw

String. When the keyword response parameter associated with the call is empty, camp_kw contains the name of the ad campaign. When the keyword contains a value, camp_kw contains the campaign name and the keyword separated by a colon (:) and a space. For example:

"Example Company National Campaign: widgets"

custom_id

The value of the Custom reference ID field on the Account Settings + Company Info and Setting page in the Call Analytics user interface.

custom1 through custom20

The value of custom field n (where n is a value of 1 through 20) for the call. A custom field is an identifier whose value can be added to a call. You can define custom fields for an account on the Account Settings + Call Log Custom Fields page in the Call Analytics user interface or by calling the `call.custom.define` method. You can assign a custom field value to a call (or calls) by using the Call Analytics user interface or by calling the `call.custom.set` method. By assigning a custom field to certain calls, you can segment data in reports to study specific types of calls.

customer_type

String. Identifies the caller as commercial, residential, or other.

Note `customer_type` is only available when reverse lookup is enabled on the account.

Note A blank or null value may be returned if data is not available.

dna_class

String. The Call DNA classification of the call, if available. Refer to Table 2, the Call DNA Classifications for more details.

duration

String. The duration of the call, represented as a string showing the hours, minutes, and seconds of the call duration in "hh:mm:ss" format.

g_name

String. The name of the ad group associated with the call.

listenedto

Number. The value is set to "1" if a user has listened to the call; otherwise, the value is set to "0".

note

String. Optional information that has been entered about a call. You can enter a note for a call (or calls) by using the Call Analytics user interface or by calling the `call.set` method.

numlookup

Number. For accounts with Reverse Address Lookup enabled, the name address, city, state, and postal code returned from a directory lookup on the caller ID. For accounts with Email Lookup enabled in Account Settings, the email returned from the directory lookup will be appended to this string.

phone_type

String. Identifies the call as landline, mobile or other.

Note `phone_type` is only available when reverse lookup is enabled on the account.

Note A blank or null value may be returned if data is not available.

redacted

Boolean. Indicates if redaction has occurred in the recording. For accounts with Call Recording Redaction enabled:

- » The value is True if there are > 0 seconds of redaction in the recording.
- » The value is False if nothing was redacted in the recording or if Call Recording Redaction is not enabled on the account.

status

String. The status of the ad campaign where the value returned is either "active" or "disabled".

high_intent

String. Indicates a call that shows high interest or high probability for conversion.

department

Boolean. Identifies which department received the call depending on how this value is configured. Example departments: sales, service, support.

agent_sentiment

String. The sentiment score of the call based on the positive or negative emotion of the agent.

caller_sentiment

Number. The sentiment score of the call based on the positive or negative emotion of the caller.

Include Call DNA

The Include Call DNA Response parameters which are returned in the result when the `include_dna` request parameter is set to true.

Note The Include Call DNA response includes the Basic and Extended properties described above in the Response section and the following additional properties, if available:

Response syntax

```
"result": [  
  {  
    .../  
    "agent_speech_duration": "string",  
    "agent_ring_count": integer,  
    "caller_speech_duration": "string",  
    ...  
  }  
]
```

Response parameters

agent_ring_count

Number. The number of rings before the call was picked up by the agent, voice mail, or other system.

agent_speech_duration

String. The total amount of time in seconds that the agent spoke during the call. The returned value is a string containing a double-precision floating point number.

caller_speech_duration

String. The total amount of time in seconds that the caller spoke during the call. The returned value is a string containing a double-precision floating point number.

Include Spotted Keywords

The "spotted" keywords for a call (that is, the keywords found in a call transcription) are returned in the result when the `include_spotted_keywords` request parameter is set to true. The Include Keyword Spotting response includes the Basic and Extended properties described above and the following additional key-value pairs:

Response syntax

```
"result": [  
  {  
    ...,  
    "agent_spotted_keywords": ["string", "string", ....],  
    "caller_spotted_keywords": ["string", "string", ...],  
    ...  
  }  
]
```

Response parameters

agent_spotted_keywords

String. A list containing the agent keywords spotted in the transcription of the call. If no agent keywords are found, the list returns (0 items).

caller_spotted_keywords

A list containing the caller keywords spotted in the transcription of the call. If no caller keywords were found, an empty list returns (0 items).

Note If no keywords were spotted on a call, both lists still return, but they will be empty (0 items).

The total number of keywords that can be returned is 300 per speaker.

Each keyword is a string that can be a single word or a phrase that contains at most three words separated by a space.

Include High Intent

Allows you to specify whether or not to return high intent information about the call; in other words, whether the call is likely to result in a conversion based on criteria you set. If the flag is set to "true", it gets the full call log for a specific account and date range, allowing for expanded refinement parameters for the additional filtering of the call log data.

If the parameter is not included or is set to false, the high intent information is not returned in the reporting results.

Response syntax

```

"result": [
  {
    "duration": "00:03:25",
    "forwardno": "2065551212",
    "inboundno": "6468351212",
    "call_status": "ANSWER",
    "call_start": "2018-07-12T20:19:15Z",
    "custom18": "",
    "call_end": "2018-07-12T20:22:40Z",
    "call_id": "Cs2WH1tHt8N89bUc",
    "keyword": "",
    "acct": "fwABAVr0XXXgrwCV",
    "custom19": "",
    "disposition": "",
    "ring_duration": 14,
    "custom11": "",
    "custom9": "",
    "note": "",
    "ass_user": "",
    "custom13": "",
    "g_name": "Automobile",
    "redacted": false,
    "status": "active",
    "custom4": "",
    "custom12": "",
    "custom2": "",
    "call_s": "2018-07-12 16:19:15",
    "cmpid": "fwABAVrXXX8qYwB6",
    "listenedto": 1,
    "numlookup": "",
    "caller_speech_duration": 46.63,
    "custom8": "",
    "answer_offset": 27,
    "a_name": "Marx MAG of Jese City - 7311",
    "caller_number": "2063311212",
    "agent_sentiment": 58.8922278973907,
    "camp_kw": "T 1 Dealer",
    "agent_ring_count": 3,
    "custom5": "",
    "custom17": "",
    "revenue": "",
    "custom15": "",
    "call_duration": 205,
    "custom3": "",
    "caller_sentiment": 19.3324938210398,
    "high_intent": "Yes",
    "custom6": "",
    "custom16": "",
  }
]

```

```

        "custom7": "",
        "department":
        {
            "Unknown": false,
            "Service": true,
            "Sales": false
        },
        "caller_name": "",
        "custom20": "",
        "custom1": "",
        "dna_class": "Conversation",
        "numLookup_p": 0,
        "custom10": "",
        "grpId": "fwABAVr1XXXXBwCi",
        "inbound_ext": "",
        "recorded": true,
        "custom14": "",
        "agent_speech_duration": 119.9,
        "billable_minutes": 4,
        "c_name": "T 1 Dealer",
        "assigned_to": "",
        "rating": "",
        "custom_id": "731"

        ...
    }
]

```

Response parameters

`include_high_intent`

Optional: Boolean; Specify true to return high intent information about the call. If the parameter is not included or is set to false, the high intent information is not returned in the reporting results.

`Include dni vars`

This is a Boolean flag that indicates whether you want the method to return the dni variables or not.

Response syntax

```

"result": [
    {
        ...
        "dni_vars":,
        "var1" : "String",
        "var2": "string",
        ...
    }
]

```



```
}  
]
```

Response parameters

dni_vars

Optional: Boolean object. The multi variable data associated with the call. This is applicable only for calls associated with dynamic-tracking ad campaigns that are configured for multi-variable collection.

Permissions

The API user must be a member of the specified account. Additionally, the method restricts the search to the calls associated with the ad groups that the API user is a member of.

Note Users with Account Admin permissions on the specified account are members of all groups in the account, and users with Client Admin permissions are members of all groups in all client accounts.

Remarks

Each call in the call log is associated with an ad campaign, which is a member of an ad group. Both the Call Analytics user interface and the call.search method restrict users to seeing only those calls that are associated with the groups the user is a member of.

Example

The example request is for all calls that started on March 23, 2015 between 9:00 AM and 5:00 PM, Pacific Daylight Time for the specified account.

Request parameters:

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "method": "call.search",  
  "params": [  
    "QrOXxEE9-fAXXXD",  
    {  
      "call_boundary": "start",  
      "start": "2015-03-23T09:00:00-7:00",  
      "end": "2015-03-23T14:17:00-7:00",  
    }  
  ]  
}
```

```

        "subacct":true
    }
]
}

```

Response parameters

The response has been truncated after the first object for readability.

Note The `call_start` and `call_end` values will return in the UTC time format.

```

[
  {
    "keyword": "",
    "call_status": "ANSWER",
    "numlookup_p": 1,
    "forwardno": "4255551212",
    "rating": "",
    "acct": "QrOXxEE9-fAXXXxD",
    "cmpid": "CA6phXXXX8RdggEP",
    "grpId": "CA6phUXXXsMdlAB6",
    "answer_offset": 4,
    "assigned_to": "",
    "inboundno": "8885551212",
    "recorded": false,
    "inbound_ext": "",
    "call_start": "2015-03-23T17:02:41Z",
    "disposition": "",
    "caller_number": "6785551212",
    "ring_duration": 4,
    "call_id": "CtjSLFXXXXzAJ9ABD",

```

```

    "revenue": "",
    "numlookup": "Hector Iribe===1539 W Mesa Dr \n===RIA-ALTO===CA===92376===iribehector@ymail.com",
    "call_duration": 11,
    "call_end": "2015-03-23T17:02:52Z",
    "caller_name": "Feld Dea"
  },
  {
    ...
  }
]

```

call.transcript.get

The Call Transcription service converts spoken English conversations into text while a call is happening. You do not need to enable call recording to get a transcript of a call. Formatted by agent and by caller, the transcript is available in the call log shortly after a call ends.

Request syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "call.transcript.get",
  "params": [
    {
      "call_id": "string"
    }
  ]
}

```

Request parameters

The request requires an object with the following key-value pair:

call_id

Required string. The unique, system-generated call ID. Use the ["Response Syntax: Basic Response" on page 253](#) method to get the call ID.

Response syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    {
      "errors": integer,
      "transcript": {
        {
          "version": integer,
          "caller": [
            {
              "duration": number,
              "phrase": "string",
              "start": number
            },
            ...
          ],
          "agent": [
            {
              "duration": number,
              "phrase": "string",
              "start": number
            },
            ...
          ]
        }
      }
    }
  }
}

```

Response parameters

The result is an object that contains the following key-value pairs:

errors

The number of errors that the Call Transcription service encountered during the conversion from speech to text.

transcript

The transcript object contains the following key-value pairs:

Field	Description
agent	An array of transcription segments representing what the receiving side of the call said.
caller	An array of transcription segments representing what the originator of the call said.
version	A number indicating the transcription version.

The *agent* and *caller* arrays contains 0 or more objects with the following key-value pairs:

Field	Description
start	The number of seconds into the call the transcription segment started.
duration	The duration, in seconds, of the segment.
phrase	The transcribed text for this segment.

Note that the phrase can contain things other than actual transcribed words:

- » Non-verbal transcriptions are enclosed in square brackets, for example: "[laughter]" or "[noise]"
- » "<unk>" is a placeholder indicating that the Transcription service detected a word, but was unable to transcribe it.
- » System warnings or errors are enclosed in curly braces.
- » "{missing_segment}" indicates a final service timeout when attempting to transcribe the segment. This segment will not be transcribed.

Permissions

The API user must belong to the same group as the ad campaign associated with the specified call. Note that users with the Account Admin permission are members of all groups in an account, and users with the Client Admin permission are members of all groups in all sub-accounts.

Remarks

The method can return the transcript for only one call at a time. To get the transcripts for multiple calls:

1. Make a request to ["Response Syntax: Basic Response" on page 253](#) for a specific time period. This returns call data, including call IDs for the specified time period.
2. Process the **call.search** response and make a list (or array) of the call IDs that you want to get transcripts for.
3. Loop through the list of call IDs, passing them one at a time to **call.transcript.get**.

Phone Numbers

This section describes the methods available to search, order, and provision phone numbers.

Method	Description
number.avail	Lists the available telephone numbers in the number pool for the specified account.
number.avail.npa	Lists up to the specified quantity of available telephone numbers in the number pool for the specified account that match the specified NPA code.
number.list	Lists the telephone numbers associated with the specified account.
number.lookup.npanxx	Looks up the geographic information of the specified NPA and NXX code combination.

Method	Description
<u>number.order</u>	Adds a new number order for the specified quantity of telephone numbers from the specified NPA and NXX code combination, for the specified account.
<u>number.order.list</u>	Lists the pending number orders for the specified account.
<u>number.order.npa</u>	Adds a new number order for the specified quantity of telephone numbers from the specified NPA code, for the specified account.
<u>number.order.rc</u>	Adds a new number order for the specified quantity of telephone numbers from the specified state and rate center, for the specified account.
<u>*number.provision.exact</u>	Adds the requested telephone numbers, if available, to the number pool for the specified account. The specified account must have Real Time Number Provisioning enabled; otherwise, an error occurs. To enable Real Time Number Provisioning, contact Client Services.
<u>*number.provision.match_npanxx</u>	Adds telephone numbers that match the specified NPA and NXX codes, if available, to the number pool for the specified account.
<u>*number.provision.nearby</u>	Matches telephone numbers that match, or are geographically as close as possible to the area defined by, the specified NPA and NXX codes and, optionally, adds matched numbers, if available, to the number pool for the specified account.
<u>*number.provision.npa</u>	Adds telephone numbers that match the specified NPA code, if available, to the number pool for the specified account.
<u>*number.provision.rc</u>	Adds telephone numbers that match the specified state and rate center, if available, to the number pool for the specified account.
<u>number.return</u>	Removes the specified telephone number from the number pool of the specified account.
<u>number.search.local.by.distance</u>	Matches available telephone numbers in the same state, up to the specified quantity, for which the rate centers are within the specified distance of the rate center identified by the specified NPA and NXX codes.
<u>number.search.npa</u>	Checks if the specified quantity of telephone numbers that match the specified NPA code are available.
<u>number.search.npa.list</u>	Matches and lists available telephone numbers, up to the specified quantity, that match the specified NPA code.
<u>number.search.npanxx.associated.list</u>	Matches and lists available telephone numbers, up to the specified quantity, that either exactly match or are in the same rate center as the specified NPA and NXX code combination.

Method	Description
<u>number.search.npanxx.list</u>	Matches and lists available telephone numbers, up to the specified quantity, that exactly match the specified NPA and NXX code combination.
<u>*number.search.prefix</u>	Get available numbers that share the prefix of a target phone number.
<u>number.search.rc</u>	Checks if the specified quantity of telephone numbers are available within the specified distance of the specified state and rate center.
<u>number.search.state</u>	Checks if the specified quantity of telephone numbers are available within the NPA codes of the specified state.

*To use the method, the specified account must have Real Time Number Provisioning enabled; otherwise, an error occurs. To enable Real Time Number Provisioning, contact Client Services.

number.avail

Lists the available telephone numbers in the number pool for the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.avail",
  "params": [
    "acc"
  ]
}
```

Request parameters

acc

Required string. The unique, system-generated account ID of the specified account.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    ""
  ]
}
```

Response parameters

result

An array of strings, each of which contains an available telephone number in the number pool.

Permissions

The user must have administrator permissions for the specified account.

Remarks

If the specified account is a client account, the method first attempts to retrieve available numbers from the number pool for the specified account. If the client account does not have its own number pool, the method then attempts to retrieve available numbers from the number pool of the parent account for the specified client account.

Example

The following JSON-RPC request and response describe a successful call to the **number.avail** method. The response returns all available numbers from the number pool for the specified account.

Note: The response has been truncated for readability.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.avail",
  "params": [
    "QrOXxEE9-fATtgAD"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "number": "2062090714"
    },
    {
      "number": "2062090715"
    },
    ...
  ]
}
```

number.avail.npa

Lists up to the specified quantity of available telephone numbers in the number pool for the specified account that match the specified NPA code.

Request syntax


```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.avail.npa",
  "params": [
    "accid",
    "npa",
    qty
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

accid

Required string. The unique, system-generated account ID of the specified account.

npa

Required string, which can contain exactly 3 characters using the following pattern: `^[2-9]{1}[0-9]{2}$`. The 3-digit Numbering Plan Area (NPA) code

qty

Required integer. The maximum number of available telephone numbers to return from the number pool.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    ""
  ]
}
```

Response parameters

result

An array of strings, each of which contains an available telephone number in the number pool that matches the specified NPA code.

Permissions

The user must have administrator permissions for the specified account.

Remarks

If the specified account is a client account, the method attempts to retrieve available numbers from the number pool of the parent account for the specified client account.

Example

The following JSON-RPC request and response describe a successful call to the **number.avail.npa** method. The response returns 5 available numbers with the specified NPA code from the number pool for the specified account.

Note The response has been truncated for readability.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.avail.npa",
  "params": [
    "QrOXxEE9-fATtgAD",
    "206",
    5
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "number": "2062090644"
    },
    {
      "number": "2062090714"
    },
    {
      "number": "2062090715"
    },
    {
      "number": "2062090716"
    },
    {
      "number": "2062090781"
    }
  ]
}
```

number.list

Lists the [call tracking numbers](#) assigned to the specified account and provides the status and other information about each number.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.list",
  "params": [
    "account_id"
  ]
}
```

```
]
}
```

Request parameters

account_id

Required string. The unique, system-generated ID identifying the account to get the number list for.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "acct": "string",
      "cmpid": "string",
      "grpId": "string",
      "number": "string",
      "status": "string"
    }
  ]
}
```

Response parameters

The result is an array that contains 0 or more objects. Each object contains the following key-value pairs:

acct

When the number is associated with a campaign, *acct* is a string containing the ID of the account that the campaign was created in. The account ID in the response can differ from the account ID specified in the request when the request account ID is a parent account and campaigns have been created in sub-accounts. When a number is not associated with a campaign, *acct* is an empty string (two double quotes).

cmpid

A string containing the unique, system-generated campaign ID of the ad campaign associated with the phone number. Note that the method returns both [static](#) and [dynamic](#) campaigns. If the number is not associated with a campaign, then *cmpid* is an empty string.

grpId

A string containing the unique, system-generated group ID of the ad group that the campaign identified in *cmpid* is a member of.

number

A string containing the phone number. For the US, Canada, and other countries in the [North American Numbering Plan](#) (NANP), the phone number contains 10 digits. For all other countries, the phone number is returned in [E.164](#) format.

status

A string containing the status of the phone number. The following table describes the values for this property:

Value	Description
active	The phone number is assigned to an ad campaign identified in the <i>cmpid</i> parameter.

Value	Description
limbo	The phone number has been removed from a static campaign within the last 30 days, and cannot yet be assigned.
reserved	The phone number is available for assignment to a static campaign. Note that if you are using dynamic tracking, the call tracking numbers for your dynamic tracking campaigns are dynamically assigned from a separate pool of numbers and are not listed with the <i>reserved</i> status.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

If you have created campaigns in sub-accounts, the numbers assigned to those campaigns are associated with the parent account. If you specify the parent account ID in the request, **number.list** responds with information about all the numbers in the parent account's number pool, including the numbers assigned to campaigns in the sub-accounts. If you specify the account ID of a sub-account in the request, **number.list** lists only the numbers assigned to campaigns in the specified sub-account.

Example

The following JSON-RPC request and response describe a successful call to the **number.list** method. The response returns information about all numbers for the specified parent account as well as for a sub-account. The number with the "reserved" status is available to be assigned to a static campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.list",
  "params": [
    "CtjSZVM67c46wwBr"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "acct": "CtjSZVM67c46wwBr",
      "cmpid": "CtjSZ1M67hpdawAy",
      "grpId": "CtjSZVM67c46wwBt",
      "number": "5025551212",
      "status": "active"
    },
    {
      "acct": "CtjSZVM67c46wwBr",
      "cmpid": "CtjSZ1M67klclADe",

```

```

        "grpId": "CtjSZ1M67klclADd",
        "number": "4255551212",
        "status": "active"
    },
    {
        "acct": "CtjSZVM68JA-ewBq",
        "cmpid": "CtjSZVM68RhAQACK",
        "grpId": "CtjSZVM68JA-ewBs",
        "number": "2065551212",
        "status": "active"
    },
    {
        "acct": "",
        "cmpid": "",
        "grpId": "",
        "number": "6065551212",
        "status": "reserved"
    }
]
}

```

number.lookup.npanxx

Looks up the geographic information of the specified NPA and NXX code combination.

Request syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.lookup.npanxx",
  "params": [
    "number"
  ]
}

```

Request parameters

number

Required string, which can contain between 6 and 10 characters. A string containing the NPA and NXX code combination.

Note The value specified for this parameter must start with six numeric characters; otherwise, an error occurs. Specify either a 6-digit combination of a Numbering Plan Area (NPA) code and Numbering Exchange (NXX) code, or a telephone number.

Response syntax

```

{
  "jsonrpc": "2.0",

```

```
"id": 1,
"result": {
  "country": "string",
  "ratecenter": "string",
  "state": "string"
}
```

Response parameters

The result object contains the following key-value pairs:

country

String. The country code of the specified NPA and NXX code combination.

ratecenter

String. The rate center name of the specified NPA and NXX code combination.

state

String. The state code of the specified NPA and NXX code combination.

Permissions

None. The API user just needs valid credentials.

Example

The following JSON-RPC request and response describe a successful call to the **number.lookup.npanxx** method. The response returns information about the country, rate center, and state for the specified NPA code and NXX code.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.lookup.npanxx",
  "params": [
    "425557"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "country": "US",
    "ratecenter": "ISSAQUAH",
    "state": "WA"
  }
}
```

number.order

Adds a new number order for the specified quantity of telephone numbers from the specified NPA and NXX code combination, for the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.order",
  "params": [
    "accid",
    "npanxx",
    qty
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

accid

Required string. The unique, system-generated account ID of the specified account.

npanxx

Required string, which can contain exactly 6 characters using the following pattern: `^[2-9]{1}[0-9]{5}$`. The 6-digit combination of a Numbering Plan Area (NPA) and Numbering Exchange (NXX) code.

qty

Required integer, which can contain a value equal to or greater than 1. The quantity of telephone numbers to be ordered.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the specified account.

Example

The following JSON-RPC request and response describe a successful call to the **number.order** method. The request orders 10 telephone numbers from the specified NPA code and NXX code for the specified account.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.order",
  "params": [
    "QrOXy0Q9KMgELQDF",
    "415222",
    10
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 2
}
```

number.order.list

Lists the pending number orders for the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.order.list",
  "params": [
    "accid"
  ]
}
```

Request parameters

accid

Required string. The unique, system-generated account ID of the specified account.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      exact,
      "npa",
      "nxx",

```



```

        "grpid",
        "order_time",
        qty,
        "ratecenter",
        "state"
    }
]
}

```

Response parameters

: Required object. An object that represents a pending number order associated with the specified account.

exact

Optional boolean. **true** if the requested telephone number must contain the specified `npa` and `nxx` values; otherwise, **false** if the requested telephone number can be within the geographic area described by the specified `npa` and `nxx` values for the pending number order.

grpid

Optional string, which can contain exactly 16 characters. The unique, system-generated order ID of the pending number order.

npa

Optional string, which can contain up to 3 characters. The Numbering Plan Area (NPA) code specified for the pending number order.

nxx

Optional string, which can contain up to 3 characters. The Numbering Exchange (NXX) code specified for the pending number order.

order_time

Optional string, that represents a valid ISO-8601 . An ISO-8601 datetime value representing the order date and time of the pending number order.

Note Time zone designators are ignored. The time zone for the account is instead used.

qty

Optional integer, which can contain a value equal to or greater than 1. The quantity of telephone numbers specified for the pending number order.

ratecenter

Optional string. The rate center of the pending number order.

result

Required array. An array of objects, each of which represents a single pending number order for the specified account.

state

Optional string. The state code of the pending number order.

Permissions

The user must have administrator permissions for the specified account.

Example

The following JSON-RPC request and response describe a successful call to the **number.order.list** method. The response returns information about all pending number orders for the specified account.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.order",
  "params": [
    "QrOXy0Q9KMgELQDF"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "order_time": "2014-03-11T16:29:30Z",
      "nxx": "",
      "order_id": "CtjSZVMfOepXFQCs",
      "exact": true,
      "ratecenter": "",
      "npa": "",
      "qty": 8,
      "state": ""
    },
    {
      "order_time": "2014-03-14T17:05:53Z",
      "nxx": "",
      "order_id": "CtjSZ1MjNvEKvgA3",
      "exact": true,
      "ratecenter": "",
      "npa": "",
      "qty": 6,
      "state": ""
    },
    {
      "order_time": "2014-03-14T17:53:10Z",
      "nxx": "",
      "order_id": "CtjSZ1MjQgYR-AD_",
      "exact": true,
      "ratecenter": "",
      "npa": "",
      "qty": 2,
      "state": ""
    },
    {
      "order_time": "2014-03-27T18:41:55Z",
      "nxx": "",
      "order_id": "CtjSZ1M0cPMMKAcB",
      "exact": true,

```

```

        "ratecenter": "ARCOLA",
        "npa": "",
        "qty": 1,
        "state": "IL"
    },
    {
        "order_time": "2014-03-27T20:30:21Z",
        "nxx": "",
        "order_id": "CtjSZVM0i10TuwCD",
        "exact": true,
        "ratecenter": "BENNETT",
        "npa": "",
        "qty": 1,
        "state": "CO"
    },
    {
        "order_time": "2014-03-27T22:28:15Z",
        "nxx": "",
        "order_id": "CtjSZ1M0pf9ZNwCa",
        "exact": true,
        "ratecenter": "SCAPPOOSE",
        "npa": "",
        "qty": 2,
        "state": "OR"
    }
]
}

```

number.order.npa

Adds a new number order for the specified quantity of telephone numbers from the specified NPA code, for the specified account.

Request syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.order.npa",
  "params": [
    "accid",
    "npa",
    exact,
    qty
  ]
}

```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

`accid`

Required string. The unique, system-generated account ID of the specified account.

`exact`

Required boolean. **true** if the requested telephone number must contain the specified `npa` value; otherwise, **false** if the requested telephone number can be within the geographic area described by the specified `npa` value.

`npa`

Required string, which can contain exactly 3 characters using the following pattern: `^[2-9]{1}[0-9]{2}$`. The 3-digit Numbering Plan Area (NPA) code of the new number order.

`qty`

Required integer, which can contain a value equal to or greater than 1. The quantity of telephone numbers to be ordered.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "result"
}
```

Response parameters

`result`

Required string. A string that contains the unique, system-generated order ID of the new number order.

Permissions

The user must have administrator permissions for the specified account.

Example

The following JSON-RPC request and response describe a successful call to the **number.order.npa** method. The request orders 10 telephone numbers from the specified NPA code for the specified account.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.order",
  "params": [
    "QrOXy0Q9KMgELQDF",
    "415",
    true,
    10
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "Cmymc1NOo7Av5QD1"
}
```

number.order.rc

Adds a new number order for the specified quantity of telephone numbers from the specified state and rate center, for the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.order.rc",
  "params": [
    "accid",
    "state",
    "ratecenter",
    exact,
    qty
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

accid

Required string. The unique, system-generated account ID of the specified account.

exact

Required boolean. **true** if the requested telephone number must contain the specified `ratecenter` value; otherwise, **false** if the requested telephone number can be within the geographic area described by the specified `ratecenter` value.

qty

Required integer, which can contain a value equal to or greater than 1. The quantity of telephone numbers to be ordered.

ratecenter

Required string. The rate center name of the new number order.

state

Required string. The state code of the new number order.

Response syntax

```
{
  "jsonrpc": "2.0",
```

```
"id": 1,  
"result": "result"}
```

Response parameters

result

Required string. A string that contains the unique, system-generated order ID of the new number order.

Permissions

The user must have administrator permissions for the specified account.

Example

The following JSON-RPC request and response describe a successful call to the **number.order.rc** method. The request orders 10 telephone numbers from the specified state and rate center for the specified account.

Request

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "method": "number.order.rc",  
  "params": [  
    "QrOXy0Q9KMgELQDF",  
    "WA",  
    "SEATTLE",  
    false,  
    10  
  ]  
}
```

Response

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "result": "Cmymc1NOo7Av5QD2"}  
}
```

number.provision.exact

Adds the requested phone numbers, if available, to the number pool of the specified account.



Note: The specified account must have Real Time Number Provisioning enabled; otherwise, an error occurs. To enable Real Time Number Provisioning, contact Client Services.

Request syntax

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "method": "number.provision.exact",  
  "params": [  
    "QrOXy0Q9KMgELQDF",  
    "WA",  
    "SEATTLE",  
    false,  
    10  
  ]  
}
```

```

    "method": "number.provision.exact",
    "params": [
      "account_id",
      [
        "string1",... "string5"
      ]
    ]
  }

```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

account_id

Required string. The unique, system-generated account ID.

numbers

Required array, which can contain between 1 and 5 items. An array of strings, each of which contains a phone number to be added to the specified account's number pool.

Response syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    "string1", .... "string5"
  ]
}

```

Response parameters

result

An array of strings, each of which contains a phone number that was successfully added to the number pool.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

The response contains only the phone numbers that were successfully provisioned. If none of the specified telephone numbers are available, an error occurs.

Example

The following JSON-RPC request and response describe a successful call to the **number.provision.exact** method. The request attempted to provision 2 phone numbers for the specified account, and the response returned the phone number that was successfully provisioned.

Request

```

{
  "jsonrpc": "2.0",
  "id": 1,

```

```

    "method": "number.provision.exact",
    "params": [
      "QrOXxEE9-fATtgAD",
      [
        "2062090714",
        "2062090715"
      ]
    ]
  }

```

Response

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    "2062090714"
  ]
}

```

number.provision.match_npanxx

Adds phone numbers that match the specified NPA and NXX codes, if available, to the number pool for the specified account.



Note: The specified account must have Real Time Number Provisioning enabled; otherwise, an error occurs. To enable Real Time Number Provisioning, contact Client Services.

Request syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.provision.match_npanxx",
  "params": [
    "account_id",
    "npa",
    "nxx",
    quantity
  ]
}

```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

account_id

Required string. The unique, system-generated account ID.

npa

Required string. The 3-digit Numbering Plan Area (NPA) code to match. The allowed numbers for the *npa* string are: [2-9] for the first digit, and [0-9] for the second and third digits. In other words, the specified *npa* string must match the following regular expression: `^[2-9]{1}[0-9]{2}$`

nxx

Required string. The 3-digit Numbering Exchange (NXX) code to match. You can specify any three numbers for the *nxx* string (`^[0-9]{3}$`).

quantity

Required integer. Specify how many phone numbers that you want added to the number pool of the specified account.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    ""
  ]
}
```

Response parameters

result

An array of strings, each of which contains a phone number that was successfully added to the number pool.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

The response contains only the phone numbers that were successfully provisioned. If no phone numbers are available, an error occurs.

Example

The following JSON-RPC request and response describe a successful call to the **number.provision.match_npanxx** method. The request attempted to provision 2 phone numbers in the specified NPA code and NXX code for the specified account, and the response returned the phone numbers that were successfully provisioned.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.provision.match_npanxx",
  "params": [
    "CtjSZVM67c46wwBr",
    "415",
    "222",
    2
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    "4152404973",
    "4155270732"
  ]
}
```

number.provision.nearby

Searches for and optionally provisions phone numbers that are geographically as close as possible to the location defined by the NPA-NXX combination.



Note: If *provision* is set to **true**, the specified account must have Real Time Number Provisioning enabled; otherwise, an error occurs. To enable Real Time Number Provisioning, contact Client Services.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.provision.nearby",
  "params": [
    "account_id",
    "npa",
    "nxx",
    quantity,
    provision
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

account_id

Required string. The unique, system-generated account ID.

npa

Required string. The 3-digit Numbering Plan Area (NPA) code to match. The allowed numbers for the *npa* string are: [2-9] for the first digit, and [0-9] for the second and third digits. In other words, the specified *npa* string must match the following regular expression: `^[2-9]{1}[0-9]{2}$`

nxx

Required string. The 3-digit Numbering Exchange (NXX) code to match. You can specify any three numbers for the *nxx* string (`^[0-9]{3}$`).

quantity

Required integer. Specify how many phone numbers that you want added to the number pool of the specified account. The maximum that you can request is 10.

provision

Required boolean. **true** to search for and add phone numbers to the number pool of the specified account; otherwise, **false** to just search for phone numbers without adding them.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "distance": integer,
      "is_exact": boolean,
      "is_local": boolean,
      "number": "string",
      "ratecenter": "string"
    }
  ]
}
```

Response parameters

The result is an array of 0 or more objects each of which contains information about a phone number that was successfully matched or added to the number pool. Each object contains the following key-value pairs:

distance

Integer. The distance, in miles, between the rate center of the specified *npa* and *nxx* and the rate center of the returned phone number. Note that *distance* can be 0.

is_exact

Boolean. **true** if the returned phone number contains the specified *npa* and *nxx*; otherwise, **false**.

is_local

Optional boolean. Indicates whether *number* is subject to local tolling when called from the specified *npa* and *nxx*. **true** if the returned phone number is local when called from the specified *npa* and *nxx*; **false** indicates a non-local toll (long distance tolling, for example).

number

Optional string. The phone number that is either an exact or nearby match of the specified *npa* and *nxx*.

ratecenter

Optional string. The rate center name of the phone number.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

Provisioned means that a phone number is added to the number pool of the specified account.

The method searches and optionally provisions numbers that are geographically as close as possible to the location defined by the combination of the specified *npa* and *nxx*. If the *provision* parameter is set to **true**, the response contains information about only the telephone numbers that were successfully provisioned; otherwise, the response contains information about the telephone numbers that could be successfully provisioned. In either case, if no telephone numbers are available, an error occurs.

Example

The following JSON-RPC request and response describe a successful call to the **number.provision.nearby** method. The request attempts to provision 2 telephone numbers as geographically close as possible to the specified NPA code and NXX code, and the response returns information about the provisioned telephone numbers.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.provision.nearby",
  "params": [
    "CtjSZVM67c46wwBr",
    "415",
    "222",
    2,
    true
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "distance": 0,
      "number": "4155707418",
      "ratecenter": "SNFC JUNPR",
      "is_exact": false,
      "is_local": true
    },
    {
      "distance": 0,
      "number": "4158900816",

```

```

        "ratecenter": "SNFC JUNPR",
        "is_exact": false,
        "is_local": true
    }
}

```

number.provision.npa

Adds telephone numbers that match the specified NPA code, if available, to the number pool for the specified account. This method allows you to provision new toll-free numbers by specifying a toll-free NPA (800, 888, 877, 866, or 855). This method *can* return numbers that are in limbo.



Note: The specified account must have Real Time Number Provisioning enabled; otherwise, an error occurs. To enable Real Time Number Provisioning, contact Client Services.

Request syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.provision.npa",
  "params": [
    "account_id",
    "npa",
    quantity
  ]
}

```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

account_id

Required string. The unique, system-generated account ID.

npa

Required string. The 3-digit Numbering Plan Area (NPA) code to match. The allowed numbers for the *npa* string are: [2-9] for the first digit, and [0-9] for the second and third digits. In other words, the specified *npa* string must match the following regular expression: `^[2-9]{1}[0-9]{2}$`

quantity

Required integer. Specify how many phone numbers that you want added to the number pool of the specified account.

Response syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,

```

```
    "result": [
        ""
    ]
}
```

Response parameters

result

An array of strings, each of which contains a phone number that was successfully added to the number pool.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

The response contains only the phone numbers that were successfully provisioned. If no phone numbers are available, an error occurs. In this case, you can try another *npa* value, or use a different ordering method.

Example

The following JSON-RPC request and response describe a successful call to the **number.provision.npa** method. The request attempted to provision 2 telephone numbers in the specified NPA code for the specified account, and the response returned the telephone numbers that were successfully provisioned.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.provision.npa",
  "params": [
    "CtjSZVM67c46wwBr",
    "415",
    2
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    "4152404973",
    "4155270732"
  ]
}
```

number.provision.rc

Adds phone numbers that match the specified state and rate center, if available, to the number pool for the specified account.



Note: The specified account must have Real Time Number Provisioning enabled; otherwise, an error occurs. To enable Real Time Number Provisioning, contact Client Services.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.provision.rc",
  "params": [
    "account_id",
    "state",
    "rate_center",
    quantity
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

account_id

Required string. The unique, system-generated account ID.

state

Required string. The state code to match.

rate_center

Required string. The rate center name to be matched.

quantity

Required integer. Specify how many phone numbers that you want added to the number pool of the specified account.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    ""
  ]
}
```

Response parameters

result

An array of strings, each of which contains a phone number that was successfully added to the number pool.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

The response contains only the phone numbers that were successfully provisioned. If no phone numbers are available, an error occurs.

Example

The following JSON-RPC request and response describe a successful call to the **number.provision.rc** method. The request attempted to provision 2 phone numbers in the specified state and rate center for the specified account, and the response returned the telephone numbers that were successfully provisioned.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.provision.rc",
  "params": [
    "CtjSZVM67c46wwBr",
    "WA",
    "SEATTLE",
    2
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    "2062090531",
    "2062090903"
  ]
}
```

number.return

Removes the specified [call tracking number](#) from the number pool of the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.return",
  "params": [
    "account_id",
    "number"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

account_id

Required string. The unique, system-generated ID that identifies the account that the number is associated with.

number

Required string. The call tracking number to be returned. For phone numbers in the US and other countries in the [North American Numbering Plan](#) (NANP), the string must contain exactly 10 digits. For other countries, the number must be in [E.164](#) format.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

If the specified number is in use by an ad campaign, an error is returned.

Example

The following JSON-RPC request and response describe a successful call to the **number.return** method. The request removes the specified phone number from the number pool of the specified account.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.return",
  "params": [
    "QrOXxEE9-fATtgAD",
    "2065551212"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
```

```
{
  "result": 2
}
```

number.search.local.by.distance

Matches available phone numbers that are available to be provisioned that are in the same state, up to the specified count, for which the rate centers are within the specified distance of the rate center identified by the specified NPA and NXX codes.

Request syntax

Required parameters are in bold.

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.search.local.by.distance",
  "params": [
    {
      "npa": "string",
      "nxx": "string",
      "count": integer,
      "distance": integer
    }
  ]
}
```

Request parameters

The *params* array requires one (and only one) object with the following key-value pairs:

npa

Required string, which can contain exactly 3 characters. The 3-digit Numbering Plan Area (NPA) code to match.

nxx

Required string, which can contain exactly 3 characters. The 3-digit Numbering Exchange (NXX) code to match.

count

Optional integer. The number of available telephone numbers to check. If not specified, the default value is 5.

distance

Optional integer. The maximum distance, in miles, from the rate center of the specified NPA and NXX code to the rate center of a telephone number to be matched. If not specified, the default value is 15.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    "string"
  ]
}
```

Response parameters

result

An array of strings, each of which contains a telephone number that was successfully matched.

Permissions

None. The API user just needs valid credentials.

Remarks

Priority is given to numbers that exactly match the specified NPA code and NXX code, followed by numbers in the same rate center. Finally, numbers in other rate centers are listed by proximity, with nearer numbers listed first.

Example

The following JSON-RPC request and response show a successful call to the **number.search.local.by.distance** method. The request searches for up to 5 telephone numbers that are within 10 miles of the specified NPA code and NXX code.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.search.local.by.distance",
  "params": [
    {
      "count": 5,
      "distance": 10,
      "npa": "415",
      "nxx": "222"
    }
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    "4155707418",
    "4158900816",
    "4158900062",
    "4158900731",
    "4158900819"
  ]
}
```

number.search.npa

Checks if the specified quantity of telephone numbers that match the specified NPA code are available.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.search.npa",
  "params": [
    "npa",
    qty
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

npa

Required string, which can contain exactly 3 characters using the following pattern: `^[2-9]{1}[0-9]{2}$`. The 3-digit Numbering Plan Area (NPA) code to check.

qty

Required integer. The quantity of available telephone numbers to check.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      available,
      "distance"
    }
  ]
}
```

Response parameters

result: Optional object. An object that represents the availability of telephone numbers in a single NPA code.

available

Optional boolean. **true** if the specified quantity of telephone numbers are available in the NPA code; otherwise, **false**.

distance

Optional string. The 3-digit Numbering Plan Area (NPA) code.

result

Required array. An array of objects, each of which represents the availability of telephone numbers in a single NPA code.

Permissions

The user must have user permissions.

Example

The following JSON-RPC request and response describe a successful call to the **number.search.npa.list** method. The request checks if 5 telephone numbers are available within the specified NPA code.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.search.npa",
  "params": [
    "415",
    5
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "npa": "415",
      "available": true
    }
  ]
}
```

number.search.npa.list

Matches and lists available telephone numbers, up to the specified quantity, that match the specified NPA code.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.search.npa.list",
  "params": [
    "npa",
    qty
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

npa

Required string, which can contain exactly 3 characters using the following pattern: `^[2-9]{1}[0-9]{2}$`. The 3-digit Numbering Plan Area (NPA) code to match.

qty

Required integer, which can contain a value between 0 and 5. The quantity of available telephone numbers to list.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    ""
  ]
}
```

Response parameters

: Required string, which can contain exactly 10 characters. A telephone number that was successfully matched.

result

Required array. An array of strings, each of which contains a telephone number that was successfully matched.

Permissions

The user must have user permissions.

Example

The following JSON-RPC request and response describe a successful call to the **number.search.npa.list** method. The request searches for up to 5 telephone numbers that are within the specified NPA code.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.search.npa.list",
  "params": [
    "415",
    5
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    "4155707418",
    "4158900816",
    "4158900062",
    "4158900731"
  ]
}
```

```
    "4158900819"  
  ]  
}
```

number.search.npanxx.associated.list

Matches and lists available telephone numbers, up to the specified quantity, that either exactly match or are in the same rate center as the specified NPA and NXX code combination.

Request syntax

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "method": "number.search.npanxx.associated.list",  
  "params": [  
    "npa",  
    "nxx",  
    qty  
  ]  
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

npa

Required string, which can contain exactly 3 characters using the following pattern: `^[2-9]{1}[0-9]{2}$`. The 3-digit Numbering Plan Area (NPA) code to match.

nxx

Required string, which can contain exactly 3 characters using the following pattern: `^[0-9]{3}$`. The 3-digit Numbering Exchange (NXX) code to match.

qty

Required integer, which can contain a value between 0 and 5. The quantity of available telephone numbers to list.

Response syntax

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "result": [  
    ""  
  ]  
}
```

Response parameters

: Required string, which can contain exactly 10 characters. A telephone number that was successfully matched.

result

Required array. An array of strings, each of which contains a telephone number that was successfully matched.

Permissions

The user must have user permissions.

Remarks

If the specified quantity of telephone numbers is not found in the specified NPA code and NXX code, the search is expanded to include telephone numbers in the same rate center as the specified NPA code and NXX code.

Example

The following JSON-RPC request and response describe a successful call to the **number.search.npanxx.associated.list** method. The request searches for 5 telephone numbers that are within the specified NPA code and NXX code, or, if the desired quantity of numbers cannot be found, within the same rate center as the specified NPAA code and NXX code.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.search.npanxx.associated.list",
  "params": [
    "415",
    "222",
    5
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    "4159062311",
    "4159063397",
    "4155237855",
    "4155237802",
    "4155270755"
  ]
}
```

number.search.npanxx.list

Matches and lists available telephone numbers, up to the specified quantity, that exactly match the specified NPA and NXX code combination.

Request syntax


```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.search.npanxx.list",
  "params": [
    "npa",
    "nxx",
    qty
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

npa

Required string, which can contain exactly 3 characters using the following pattern: `^[2-9]{1}[0-9]{2}$`. The 3-digit Numbering Plan Area (NPA) code to match.

nxx

Required string, which can contain exactly 3 characters using the following pattern: `^[0-9]{3}$`. The 3-digit Numbering Exchange (NXX) code to match.

qty

Required integer, which can contain a value between 0 and 5. The quantity of available telephone numbers to list.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    ""
  ]
}
```

Response parameters

: Required string, which can contain exactly 10 characters. A telephone number that was successfully matched.

result

Required array. An array of strings, each of which contains a telephone number that was successfully matched.

Permissions

The user must have user permissions.

Example

The following JSON-RPC request and response describe a successful call to the **number.search.npanxx.list** method. The request searches for 5 telephone numbers that are within the specified NPA code and NXX code.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.search.npanxx.list",
  "params": [
    "415",
    "222",
    5
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    "4152220111",
    "4152220125"
  ]
}
```

number.search.prefix

Returns an array of available phone numbers that match the beginning of the specified phone number or partial phone number (the prefix). Use the *match_length* parameter to specify the number of digits that must exactly match in the specified phone number or prefix. To make the phone numbers available for use in ad campaigns, set the *provision* parameter to **true** and specify a quantity from 1 to 10.



Note: The specified account must have Real Time Number Provisioning enabled; otherwise, an error occurs. To enable Real Time Number Provisioning, contact Client Services.

Request syntax

Required parameters are in bold.

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.search.prefix",
  "params": [
    {
      "account_id": "string",
      "phone": "string",
      "match_length": integer,
      "quantity": integer,
      "provision": boolean,
      "days_clean": integer,
      "allow_limbo": "string"
    }
  ]
}
```

```

    }
  ]
}

```

Request parameters

The *params* array requires one (and only one) object with the following key-value pairs:

account_id

Required string. The unique, system-generated account ID. An account is associated with one country. The country code that you specify in the *phone* parameter must match the country configured in the account.

phone

Required string. The full or partial phone number in E.164 format to use in the search. For example, to find US phone numbers (+1 country code) in the 206 area code with a NXX of 745, specify "+1206745" and specify 7 in the *match_length* parameter, which is the number of digits specified in the *phone* parameter.

match_length

Optional integer. The number of digits that have to exactly match the digits in the beginning of the *phone* parameter. For the US, Canada, and other countries in the North American Numbering Plan (NANP), the default value and the minimum value that you can specify is 2. For all other countries, the default value and the minimum value that you can specify is the length of the country code, which is typically either 2 or 3.

quantity

Optional integer, which defaults to 10. The value you specify must be between 1 and 10. To provision phone numbers, you must include the *quantity* parameter in the request.

provision

Optional boolean, which defaults to **false**. If **false**, the phone numbers returned from the search are numbers that could be used for ad campaigns, but the numbers are not provisioned. That is, the numbers are not made available in the account's number pool and cannot be assigned to ad campaigns. If **true**, the returned phone numbers will be provisioned if you specify a valid value in the *quantity* parameter.

Note The entire quantity must be available for any numbers to be provisioned.

days_clean

Optional integer, which defaults to the account's specified clean period. The number of days since the phone number was last used in an ad campaign.

allow_limbo

Optional string. Specify your account ID if you want to include numbers from the account's number pool that have been clean for less than 30 days (referred to as the limbo state). Specify **"1"** if you want to include limbo-state numbers from the global number pool and from the account's number pool.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    matching number 1,
    matching number 2,
    ...
    matching number n
  ]
}
```

Response parameters

result

Contains an array of phone numbers as strings in E.164 format that match the target phone number or prefix up to the specified `match_length`.

Permissions

The API user must have Account Admin permissions in the account specified in the `account_id` parameter. If you specify an account ID in the `allow_limbo` parameter, the API user must also have Account Admin permissions in that account.

Remarks

You can use the `phone` and `match_length` parameters to get your first choice of phone numbers, and similar ones if your first choice is not available. For example, if you prefer a phone number in inner London (inner London phone numbers start with the 20 London dial code and are typically followed by 7), but will accept any London phone number, specify "+44207" and set `match_length` to 4, which is the number of digits in the UK country code + the London dial code.

Example

The following JSON-RPC request and response describe a successful call to the **number.search.prefix** method. There is a 7-digit prefix in the `phone` parameter and `match_length` is set to 3. This indicates that the first choice would be for phone numbers in the 425 area code with NXX of 214, but any phone number in the 425 area code will be acceptable. Because the `provision` parameter was not included in the request, the phone numbers in the response are not provisioned.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.search.prefix",
  "params": [
    {
      "account_id": "QrOXxEE9-fATtgAD",
      "phone": "+1425214",
      "match_length": 4,
      "quantity": 3,
    }
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    "+1425212xxxx",
    "+1425214xxxx",
    "+1425214xxxx"
  ]
}
```

number.search.rc

Checks if the specified quantity of telephone numbers are available within the specified distance of the specified state and rate center.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.search.rc",
  "params": [
    "state",
    "rc",
    dist,
    qty
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

dist

Required integer, which can contain a value equal to or greater than 0. The additional distance, in miles, to check from the specified state and rate center. Set the value of this parameter to 0 to check only the specified state and rate center.

qty

Required integer. The quantity of available telephone numbers to check.

rc

Required string, which can contain between 1 and 16 characters. The rate center name of the specified state and rate center.

state

Required string, which can contain between 1 and 4 characters. The state code of the specified state and rate center.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      available,
      "dis",
      "rc",
      "state"
    }
  ]
}
```

Response parameters

: Optional object. An object that represents the availability of telephone numbers in the resulting state and rate center.

available

Optional boolean. **true** if the specified quantity of telephone numbers are available in the resulting state and rate center; otherwise, **false**.

dis

Optional string. The distance, in miles, between the resulting state and rate center and the specified state and rate center.

rc

Optional string. The rate center name of the resulting state and rate center.

result

Required array. An array of objects, each of which represents the availability of telephone numbers within a resulting state and rate center.

state

Optional string. The state code of the resulting state and rate center.

Permissions

The user must have user permissions.

Remarks

Priority is given to the specified state and rate center, followed by states and rate centers within the specified radius, listed by proximity, with nearer rate centers listed first.

Example

The following JSON-RPC request and response describe a successful call to the **number.search.rc** method. The request checks if 5 telephone numbers are available either within the specified state and rate center, or in other states and rate centers within a 10 mile radius of the specified state and rate center.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.search.rc",
  "params": [
    "WA",
    "SEATTLE",
  ]
}
```

```
        10,  
        5  
    ]  
}
```

Response

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "result": [  
    {  
      "rc": "SEATTLE",  
      "dist": 0,  
      "available": true,  
      "state": "WA"    },  
    {  
      "rc": "BELLEVUE",  
      "dist": 5,  
      "available": true,  
      "state": "WA"    },  
    {  
      "rc": "KIRKLAND",  
      "dist": 7,  
      "available": true,  
      "state": "WA"    },  
    {  
      "rc": "BAINBDG IS",  
      "dist": 8,  
      "available": true,  
      "state": "WA"    }  
  ]  
}
```

number.search.state

Checks if the specified quantity of telephone numbers are available within the NPA codes of the specified state.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.search.state",
  "params": [
    "state",
    qty
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

qty

Required integer. The quantity of available telephone numbers to check.

state

Required string, which can contain up to 4 characters. The state code of the specified state.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      available,
      "distance"
    }
  ]
}
```

Response parameters

: Optional object. An object that represents the availability of telephone numbers in a single NPA code.

available

Optional boolean. **true** if the specified quantity of telephone numbers are available in the NPA code; otherwise, **false**.

distance

Optional string. The 3-digit Numbering Plan Area (NPA) code.

result

Required array. An array of objects, each of which represents the availability of telephone numbers in a single NPA code for the specified state.

Permissions

The user must have user permissions.

Example

The following JSON-RPC request and response describe a successful call to the **number.search.state** method. The request checks if 5 telephone numbers are available in the specified state.

Request


```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "number.search.state",
  "params": [
    "WA",
    5
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "npa": "206",
      "available": true
    },
    {
      "npa": "253",
      "available": true
    },
    {
      "npa": "360",
      "available": true
    },
    {
      "npa": "425",
      "available": true
    },
    {
      "npa": "509",
      "available": true
    },
    {
      "npa": "564",
      "available": false
    }
  ]
}
```

Playfiles

Playfiles play a greeting to callers to an ad campaign. Playfiles customize the caller experience and provide callers with details about the business or service they have reached. Whether you upload playfiles via the API or the UI, playfiles are associated with an account, and you can set the same playfile on more than one campaign in the account.

Playfile Methods at the Account Level

The following playfile methods operate at the **account** level. Although the methods start with "ad" and they take a campaign ID in the request, these methods look up the account that the specified campaign was created in and then do the action—create a new playfile in the account, list the playfiles in the account, or delete the playfile from the account. See [Playfile Methods at the Ad Campaign Level](#) for a list of methods that you can use to set or remove ("unset") playfiles on ad campaigns.

By default, if you create playfiles in a parent account, they are not visible in sub-accounts. If you want to use the same playfile in campaigns created in sub-accounts, contact Client Services to have them enable the "Share Playfiles with Children" setting on the parent account.

Method	Description
ad.play.file.new	Creates a new playfile in the account that the specified ad campaign was created in. The playfile can then be set on the specified ad campaign—or any campaign in the account—via one of the ad.play.*.set methods or via the user interface.
ad.play.file.list	Lists the playfiles in the account that the specified ad campaign was created in.

Playfile Methods at the Ad Campaign Level

The following playfile methods allow you to control the greetings that are played to callers. Note that the "unset" methods remove the playfile from the ad campaign, but they do not delete the playfile itself. To delete a playfile you must first remove ("unset") the playfile from all campaigns and then call **ad.play.file.del** to delete the playfile.

Method	Description
ad.play.afterhr_exten.-set	Sets the playfile as the extension voice prompt for incoming calls received after business hours on the specified ad campaign.
ad.play.afterhr_exten.unset	Removes the playfile as the extension voice prompt for incoming calls received after business hours on the specified ad campaign.
ad.play.afterhrs.set	Sets the playfile as the after hours message for incoming calls received after business hours on the specified ad campaign.
ad.play.afterhrs.unset	Removes the playfile as the after hours message for incoming calls received after business hours on the specified ad campaign.
ad.play.busnhr_exten.-set	Sets the playfile as the extension voice prompt for incoming calls received during business hours on the specified ad campaign.
ad.play.busnhr_exten.unset	Removes the playfile as the extension voice prompt for incoming calls received during business hours on the specified ad campaign.
ad.play.set	Sets the playfile as the greeting for incoming calls received during business hours on the specified ad campaign.
ad.play.unset	Removes the playfile used by the specified ad campaign as the greeting for incoming calls received during business hours.

Method	Description
<u>ad.play.vmail.set</u>	Sets the playfile as the voicemail prompt for incoming calls on the specified ad campaign.
<u>ad.play.vmail.unset</u>	Removes the playfile as the voicemail prompt for incoming calls on the specified ad campaign.
<u>ad.play.whisper.set</u>	Sets the playfile as the whisper tone heard by recipients of an incoming call on the specified ad campaign.
<u>ad.play.whisper.unset</u>	Removes the playfile as the whisper tone heard by recipients of an incoming call on the specified ad campaign.

ad.play.file.new

Creates a new playfile in the account that the specified ad campaign was created in. The playfile can then be set on the specified ad campaign—or any campaign in the account—via one of the [ad.play.*.set](#) methods or via the user interface.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.file.new",
  "params": [
    "campaign_id",
    "playfile_name",
    "wav_data"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

campaign_id

Required string. The unique, system-generated campaign ID. Call Analytics looks up the account that the campaign was created in and creates the playfile in the account.

playfile_name

Required string, which can contain up to 32 characters. The name of the new playfile. The string you specify here is displayed in the user interface on the Account > Playfile List page and in the various Playfile drop-down lists on the Campaign Settings page of any ad campaign in the account.

wav_data

Required string. A Base64-encoded string containing the audio data, in Windows Audio Video (.wav) format, for the new playfile.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "descr": "string",
    "fid": "string"
  }
}
```

Response parameters

The result object contains the following key-value pairs:

descr

A string containing the name of the playfile that you specified in the request.

fid

A string containing the unique, system-generated file ID of the playfile. You specify this file ID in the requests to the [ad.play.*.set](#) methods.

Permissions

The API user must have Account Admin permissions in the account that the specified ad campaign is associated with.

Remarks

Call Analytics decodes the Base64-encoded *wav_data* string back to the original .wav file format. Before it is stored with the other playfiles, the file is transcoded to another .wav file with the following format: 8-kHz sample rate * 16-bit sample size * 1 channel. If the file cannot be converted, an error is returned. For example, if the original .wav file were a "CD-quality" recording, it would be: 44-kHz sample rate * 24bit * 2 channels (which is a really large file). Call Analytics transcodes this file to the format that playfiles are stored in, which is more appropriate (smaller file size and good audio quality) for use as a greeting in a telephone application.

By default, if you create playfiles in a parent account, they are not visible in sub-accounts. If you want to use the same playfile in campaigns created in sub-accounts, contact Client Services to have them enable the "Share Playfiles with Children" setting on the parent account.

Example Code

For details on reading a .wav file and converting it to a Base64-encoded string, see the [ad-play-file-new.php](#) and [ad-play-file-new.py](#) example code available for download from ["Example Code for JSON-RPC API" on page 34](#).

Example

The following JSON-RPC request and response show a successful call to the **ad.play.file.new** method. The request creates a new playfile named "after hours greeting" that can be set on any campaign in the account that the campaign specified in the request was created in.

Note The value of the *wav_data* parameter in the request has been truncated for readability purposes.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.file.new",
  "params": [
```

```

        "QrOXy0RPy_gATgEB",
        "after hours greeting",
        "UklGRoglAABXQVZFZm10IBQAAAAxAAEARKwAAP0iAAB ... "
    ]
}

```

Response

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "fid": "CmymclNOpCsv5wAW",
    "descr": "after hours greeting"
  }
}

```

ad.play.file.list

Lists the playfiles in the account that the specified ad campaign was created in.

Request syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.file.list",
  "params": [
    "campaign_id"
  ]
}

```

Request parameters

campaign_id

Required string. The unique, system-generated campaign ID. Call Analytics looks up the account that the campaign was created in and finds all of the playfiles in the account.

Response syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "descr": "string",
      "fid": "string"
    },
    ...
    {

```

```

        "descr": "string",
        "fid": "string"
      }
    ]
  }

```

Response parameters

The result is an array that contains 0 or more objects. Each object contains the following key-value pairs:

descr

A string containing the name of the playfile.

fid

A string containing the unique, system-generated file ID of the playfile.

Permissions

The API user must have Account Admin permissions in the account that the specified ad campaign is associated with.

Example

The following JSON-RPC request and response show a successful call to the **ad.play.file.list** method. The response returns an array of objects, each of which contains the file ID and name of a playfile in the account that the campaign specified in the request was created in.

Request

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.file.list",
  "params": [
    "QrOXy0RPy_gATgEB"
  ]
}

```

Response

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "fid": "CnymclNOpCsv5wAW",
      "descr": "after hours"
    },
    {
      "fid": "CnymclNOpCsv5wAX",
      "descr": "open hours"
    },
    {
      "fid": "CnymclNOpCsv5wAY",
      "descr": "sales dept greeting"
    }
  ]
}

```

```
}  
]  
}
```

ad.play.afterhr_exten.set

Sets the specified playfile as the extension voice prompt for incoming calls received after business hours on the specified ad campaign.

Request syntax

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "method": "ad.play.afterhr_exten.set",  
  "params": [  
    "campaign_id",  
    "playfile_id"  
  ]  
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

campaign_id

Required string. The unique, system-generated ID of the ad campaign. You can use ["ad.list" on page 169](#) or ["ad.list.all" on page 164](#) to get the *campaign_id*.

playfile_id

Required string. The unique, system-generated file ID of the playfile to be used by the specified ad campaign as the extension voice prompt for incoming calls received after business hours. You can use ["ad.play.file.list" on page 317](#) to get a list of available playfile names and IDs.

Response syntax

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "result": integer  
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must be a Group Admin of the group that the specified campaign belongs to. Note that users with the Account Admin permission are Group Admins of all groups in an account, and parent account users with the Client Admin permission are Group Admins of all groups in all sub-accounts.

Remarks

The playfile to be set must have been previously uploaded to the account that the specified ad campaign was created in. You can use ["ad.play.file.new"](#) on [page 315](#) or the user interface to upload a playfile.

The method returns successfully even when [Extensions](#) are not enabled on the specified ad campaign. You must enable Extensions on the campaign for the playfile to be played to callers.

Example

The following JSON-RPC request and response show a successful call to the **ad.play.afterhr_exten.set** method. The request sets the After Hours Ask Extension playfile for the specified ad campaign (the first parameter) to the specified playfile (the second parameter).

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.afterhr_exten.set",
  "params": [
    "QrOXy0RPY_gATgEB",
    "QrOXzUGPqvZZAAAE"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.play.afterhr_exten.unset

Removes the playfile as the extension voice prompt for incoming calls received after business hours from the specified ad campaign.

Request syntax


```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.afterhr_exten.unset",
  "params": [
    "campaign_id"
  ]
}
```

Request parameters

campaign_id

Required string. The unique, system-generated ID of the ad campaign. You can use ["ad.list" on page 169](#) or ["ad.list.all" on page 164](#) to get the *campaign_id*.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must be a Group Admin of the group that the specified campaign belongs to. Note that users with the Account Admin permission are Group Admins of all groups in an account, and parent account users with the Client Admin permission are Group Admins of all groups in all sub-accounts.

Remarks

While the playfile reference is removed from the specified ad campaign, the playfile itself is not deleted. To delete the playfile after all references to the playfile have been removed from all ad campaigns, use `ad.play.file.del`.

Example

The following JSON-RPC request and response show a successful call to the **ad.play.afterhr_exten.unset** method. The request removes the After Hours Ask Extension playfile reference for the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.afterhr_exten.unset",
  "params": [
    "QrOXy0RPy_gATgEB"
  ]
}
```

```
}  
}
```

Response

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "result": 1  
}
```

ad.play.afterhrs.set

Sets the specified playfile as the after hours message for incoming calls received after business hours on the specified ad campaign.

Request syntax

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "method": "ad.play.afterhrs.set",  
  "params": [  
    "campaign_id",  
    "playfile_id"  
  ]  
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

campaign_id

Required string. The unique, system-generated ID of the ad campaign. You can use ["ad.list" on page 169](#) or ["ad.list.all" on page 164](#) to get the *campaign_id*.

playfile_id

Required string. The unique, system-generated file ID of the playfile to be used by the specified ad campaign as the greeting played to incoming calls received after hours. You can use ["ad.play.file.list" on page 317](#) to get a list of available playfile names and IDs.

Response syntax

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "result": integer  
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must be a Group Admin of the group that the specified campaign belongs to. Note that users with the Account Admin permission are Group Admins of all groups in an account, and parent account users with the Client Admin permission are Group Admins of all groups in all sub-accounts.

Remarks

The playfile to be set must have been previously uploaded to the account that the specified ad campaign was created in. You can use ["ad.play.file.new" on page 315](#) or the user interface to upload a playfile.

Example

The following JSON-RPC request and response show a successful call to the **ad.play.afterhrs.set** method. The request sets the After Hours playfile for the specified ad campaign (the first parameter) to the specified playfile (the second parameter).

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.afterhrs.set",
  "params": [
    "QrOXy0RPy_gATgEB",
    "QrOXzUGPqvZZAAAF"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.play.afterhrs.unset

Removes the playfile used as the after hours message for incoming calls received after business hours from the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.afterhrs.unset",
  "params": [
```

```
    "campaign_id"
  ]
}
```

Request parameters

campaign_id

Required string. The unique, system-generated ID of the ad campaign. You can use ["ad.list" on page 169](#) or ["ad.list.all" on page 164](#) to get the *campaign_id*.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must be a Group Admin of the group that the specified campaign belongs to. Note that users with the Account Admin permission are Group Admins of all groups in an account, and parent account users with the Client Admin permission are Group Admins of all groups in all sub-accounts.

Remarks

While the playfile reference is removed from the specified ad campaign, the playfile itself is not deleted. To delete the playfile after all references to the playfile have been removed from all ad campaigns, use `ad.play.file.del`.

Example

The following JSON-RPC request and response show a successful call to the `ad.play.afterhrs.unset` method. The request removes the After Hours playfile on the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.afterhrs.unset",
  "params": [
    "QrOXy0RPy_gATgEB"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.play.busnhr_exten.set

Sets the specified playfile as the extension voice prompt for incoming calls received during business hours on the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.busnhr_exten.set",
  "params": [
    "campaign_id",
    "playfile_id"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

campaign_id

Required string. The unique, system-generated ID of the ad campaign. You can use ["ad.list" on page 169](#) or ["ad.list.all" on page 164](#) to get the *campaign_id*.

playfile_id

Required string. The unique, system-generated file ID of the playfile to be used by the specified ad campaign as the extension voice prompt for incoming calls received during business hours. You can use ["ad.play.file.list" on page 317](#) to get a list of available playfile names and IDs.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must be a Group Admin of the group that the specified campaign belongs to. Note that users with the Account Admin permission are Group Admins of all groups in an account, and parent account users with the Client Admin permission are Group Admins of all groups in all sub-accounts.

Remarks

The playfile to be set must have been previously uploaded to the account that the specified ad campaign was created in. You can use ["ad.play.file.new" on page 315](#) or the user interface to upload a playfile.

The method returns successfully even when [Extensions](#) are not enabled on the specified ad campaign. You must enable Extensions on the campaign for the playfile to be played to callers.

Example

The following JSON-RPC request and response show a successful call to the **ad.play.busnhr_exten.set** method. The request sets the Business Hours Ask Extension playfile for the specified ad campaign (the first parameter) to the specified playfile (the second parameter).

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.busnhr_exten.set",
  "params": [
    "QrOXy0RPy_gATgEB",
    "QrOXzUGPqvZZAAAB"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.play.busnhr_exten.unset

Removes the playfile used as the extension voice prompt for incoming calls received during business hours from the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.busnhr_exten.unset",
  "params": [
    "campaign_id"
  ]
}
```

Request parameters

campaign_id

Required string. The unique, system-generated ID of the ad campaign. You can use ["ad.list" on page 169](#) or ["ad.list.all" on page 164](#) to get the *campaign_id*.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must be a Group Admin of the group that the specified campaign belongs to. Note that users with the Account Admin permission are Group Admins of all groups in an account, and parent account users with the Client Admin permission are Group Admins of all groups in all sub-accounts.

Remarks

While the playfile reference is removed from the specified ad campaign, the playfile itself is not deleted. To delete the playfile after all references to the playfile have been removed from all ad campaigns, use `ad.play.file.del`.

Example

The following JSON-RPC request and response show a successful call to the **ad.play.busnhr_exten.unset** method. The request removes the Business Hours Ask Extension playfile reference for the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.busnhr_exten.unset",
  "params": [
    "QrOXy0RPy_gATgEB"
  ]
}
```

```
}  
}
```

Response

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "result": 1  
}
```

ad.play.set

Sets the specified playfile as the greeting for incoming calls received during business hours on the specified ad campaign.

Request syntax

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "method": "ad.play.set",  
  "params": [  
    "campaign_id",  
    "playfile_id"  
  ]  
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

campaign_id

Required string. The unique, system-generated ID of the ad campaign. You can use ["ad.list" on page 169](#) or ["ad.list.all" on page 164](#) to get the *campaign_id*.

playfile_id

Required string. The unique, system-generated file ID of the playfile to be used by the specified ad campaign as the greeting during business hours. You can use ["ad.play.file.list" on page 317](#) to get a list of available playfile names and IDs.

Response syntax

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "result": integer  
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must be a Group Admin of the group that the specified campaign belongs to. Note that users with the Account Admin permission are Group Admins of all groups in an account, and parent account users with the Client Admin permission are Group Admins of all groups in all sub-accounts.

Remarks

The playfile to be set must have been previously uploaded to the account that the specified ad campaign was created in. You can use ["ad.play.file.new" on page 315](#) or the user interface to upload a playfile.

Example

The following JSON-RPC request and response show a successful call to the **ad.play.set** method. The request sets the Business Hours playfile for the specified ad campaign (the first parameter) to the specified playfile (the second parameter).

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.set",
  "params": [
    "QrOXy0RPy_gATgEB",
    "QrOXzUGPqvZZAAAC"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.play.unset

Removes the playfile used as the greeting for incoming calls received during business hours from the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.unset",
  "params": [
```

```
    "campaign_id"
  ]
}
```

Request parameters

campaign_id

Required string. The unique, system-generated ID of the ad campaign. You can use ["ad.list" on page 169](#) or ["ad.list.all" on page 164](#) to get the *campaign_id*.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must be a Group Admin of the group that the specified campaign belongs to. Note that users with the Account Admin permission are Group Admins of all groups in an account, and parent account users with the Client Admin permission are Group Admins of all groups in all sub-accounts.

Remarks

While the playfile reference is removed from the specified ad campaign, the playfile itself is not deleted. To delete the playfile after all references to the playfile have been removed from all ad campaigns, use `ad.play.file.del`.

Example

The following JSON-RPC request and response show a successful call to the **ad.play.unset** method. The request removes the Business Hours playfile reference for the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.unset",
  "params": [
    "QrOXy0RPY_gATgEB"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.play.vmail.set

Sets the specified playfile as the voicemail greeting for incoming calls on the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.vmail.set",
  "params": [
    "campaign_id",
    "playfile_id"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

campaign_id

Required string. The unique, system-generated ID of the ad campaign. You can use ["ad.list" on page 169](#) or ["ad.list.all" on page 164](#) to get the *campaign_id*.

playfile_id

Required string. The unique, system-generated file ID of the playfile to be used by the specified ad campaign as the voicemail prompt for incoming calls. You can use ["ad.play.file.list" on page 317](#) to get a list of available playfile names and IDs.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must be a Group Admin of the group that the specified campaign belongs to. Note that users with the Account Admin permission are Group Admins of all groups in an account, and parent account users with the Client Admin permission are Group Admins of all groups in all sub-accounts.

Remarks

The playfile to be set must have been previously uploaded to the account that the specified ad campaign was created in. You can use ["ad.play.file.new"](#) on [page 315](#) or the user interface to upload a playfile.

The method returns successfully even when [Voicemail](#) is not enabled on the specified ad campaign. You must enable Voicemail on the campaign for the playfile to be played to callers.

Example

The following JSON-RPC request and response show a successful call to the **ad.play.vmail.set** method. The request sets the Voicemail playfile for the specified ad campaign (the first parameter) to the specified playfile (the second parameter).

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.vmail.set",
  "params": [
    "QrOXy0RPy_gATgEB",
    "QrOXzUGPqvZZAAAF"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.play.vmail.unset

Removes the playfile used as the voicemail prompt for incoming calls from the specified ad campaign .

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.vmail.unset",
  "params": [
    "campaign_id"
  ]
}
```

Request parameters

campaign_id

Required string. The unique, system-generated ID of the ad campaign. You can use "ad.list" on page 169 or "ad.list.all" on page 164 to get the *campaign_id*.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must be a Group Admin of the group that the specified campaign belongs to. Note that users with the Account Admin permission are Group Admins of all groups in an account, and parent account users with the Client Admin permission are Group Admins of all groups in all sub-accounts.

Remarks

While the playfile reference is removed from the specified ad campaign, the playfile itself is not deleted. To delete the playfile after all references to the playfile have been removed from all ad campaigns, use `ad.play.file.del`.

Example

The following JSON-RPC request and response show a successful call to the `ad.play.vmail.unset` method. The request removes the Voicemail playfile reference for the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.vmail.unset",
  "params": [
    "QrOXy0RPy_gATgEB"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.play.whisper.set

Sets the specified playfile as the whisper tone heard by recipients of an incoming call on the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.whisper.set",
  "params": [
    "campaign_id",
    "playfile_id"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

campaign_id

Required string. The unique, system-generated ID of the ad campaign. You can use ["ad.list" on page 169](#) or ["ad.list.all" on page 164](#) to get the *campaign_id*.

playfile_id

Required string. The unique, system-generated file ID of the playfile to be used by the specified ad campaign as the whisper tone heard by recipients of an incoming call. You can use ["ad.play.file.list" on page 317](#) to get a list of available playfile names and IDs.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must be a Group Admin of the group that the specified campaign belongs to. Note that users with the Account Admin permission are Group Admins of all groups in an account, and parent account users with the Client Admin permission are Group Admins of all groups in all sub-accounts.

Remarks

The playfile to be set must have been previously uploaded to the account that the specified ad campaign was created in. You can use ["ad.play.file.new" on page 315](#) or the user interface to upload a playfile.

Example

The following JSON-RPC request and response show a successful call to the **ad.play.whisper.set** method. The request sets the Whisper playfile for the specified ad campaign to the specified playfile.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.whisper.set",
  "params": [
    "QrOXy0RPY_gATgEB",
    "QrOXzUGPqvZZAAAD"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

ad.play.whisper.unset

Unsets the playfile used as the whisper tone heard by recipients of an incoming call from the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.whisper.unset",
  "params": [
    "campaign_id"
  ]
}
```

Request parameters

campaign_id

Required string. The unique, system-generated ID of the ad campaign. You can use **"ad.list"** on [page 169](#) or **"ad.list.all"** on [page 164](#) to get the *campaign_id*.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must be a Group Admin of the group that the specified campaign belongs to. Note that users with the Account Admin permission are Group Admins of all groups in an account, and parent account users with the Client Admin permission are Group Admins of all groups in all sub-accounts.

Remarks

While the playfile reference is removed from the specified ad campaign, the playfile itself is not deleted. To delete the playfile after all references to the playfile have been removed from all ad campaigns, use `ad.play.file.del`.

Example

The following JSON-RPC request and response show a successful call to the `ad.play.whisper.unset` method. The request removes the Whisper playfile reference for the specified ad campaign.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.play.whisper.unset",
  "params": [
    "QrOXy0RPy_gATgEB"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 1
}
```

Postbacks

The following methods allow you to get and set postback URLs.

Method	Description
<u>ad.call_postback.get</u>	Gets the call end postback URL for the specified ad campaign.
<u>ad.call_postback.set</u>	Sets the call end postback URL for the specified ad campaign.
<u>ad.precall_postback.get</u>	Gets the call start postback URL for the specified ad campaign.
<u>ad.precall_postback.set</u>	Sets the call start postback URL for the specified ad campaign.

ad.call_postback.get

Gets the Call End Postback URL for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.call_postback.get",
  "params": [
    "campaign_id"
  ]
}
```

Request parameters

campaign_id

Required string. The unique, system-generated ID that identifies an ad campaign.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "string"
}
```

Response parameters

result

A string containing the Call End Postback URL for the specified ad campaign.

Permissions

The API user must belong to the same group as the specified ad campaign. Note that users with the Account Admin permission are members of all groups in an account, and users with the Client Admin permission are members of all groups in all client accounts.

Example

The following JSON-RPC request and response describe a successful call to the **ad.call_postback.get** method. The response returns the call end postback URL, "http://mydomain.com/call_postback", for the ad campaign specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.call_postback.get",
  "params": [
    "CtjSZ1G1CNZF0AAm"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "http://mydomain.com/call_postback"
}
```

ad.call_postback.set

Sets the Call End Postback URL for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.call_postback.set",
  "params": [
    "campaign_id",
    "call_end_postback"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

campaign_id

Required string. The unique, system-generated ID that identifies an ad campaign.

call_end_postback

Required string, which can contain up to 256 characters. The Call End Postback URL for the specified ad campaign. If the campaign already has a URL, it is replaced with the specified URL. You can specify an empty string (two double quotes) to delete a URL. See [RFC1738: Uniform Resource Locators \(URL\)](#) for detailed information on the characters safe to use in URLs.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must be a Group Admin of the group that the specified campaign belongs to. Note that users with the Account Admin permission are Group Admins of all groups in an account, and parent account users with the Client Admin permission are Group Admins of all groups in all sub-accounts.

Example

The following JSON-RPC request and response describe a successful call to the **ad.call_postback.set** method. The request sets the call end postback URL for the specified ad campaign to "http://mydomain.com/call_postback".

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.call_postback.set",
  "params": [
    "CtjSZlGlCNZF0AAm",
    "http://mydomain.com/call_postback"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

ad.precall_postback.get

Gets the call start postback URL for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
```

```
{
  "method": "ad.precall_postback.get",
  "params": [
    "cmpid"
  ]
}
```

Request parameters

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "result"
}
```

Response parameters

result

Required string. The call start postback URL for the specified ad campaign.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.precall_postback.get** method. The response returns the precall postback URL, "http://mydomain.com/precall_postback", for the ad campaign specified in the request.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.precall_postback.get",
  "params": [
    "QrOXy0RPy_gATgEB"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "http://mydomain.com/precall_postback"
}
```

ad.precall_postback.set

Sets the call start postback URL for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.precall_postback.set",
  "params": [
    "cmpid",
    "precall_postback"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

cmpid

Required string. The unique, system-generated campaign ID of the specified ad campaign.

precall_postback

Required string, which can contain up to 256 characters that represents a valid URI. The call start postback URL for the specified ad campaign.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": result
}
```

Response parameters

result

Required integer. An integer that indicates successful completion of the method, for methods that do not have explicit return values. If the method was successful, this result value is returned, and can be ignored. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The user must have administrator permissions for the account to which the specified ad campaign is associated.

Example

The following JSON-RPC request and response describe a successful call to the **ad.precall_postback.set** method. The request sets the precall postback URL for the specified ad campaign to "http://mydomain.com/precall_postback".

Request

```
{
  "jsonrpc": "2.0",
```

```
{
  "id": 1,
  "method": "ad.precall_postback.set",
  "params": [
    "CtjSZlGlCNZF0AAm",
    "http://mydomain.com/precall_postback"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

ad.calldna_postback.get

Gets the Call DNA Ready Postback URL for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.calldna_postback.get",
  "params": [
    "campaign_id"
  ]
}
```

Request parameters

campaign_id

Required string. The unique, system-generated ID that identifies an ad campaign.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "string"
}
```

Response parameters

result

A string containing the Call DNA Ready Postback end point URL for the specified ad campaign.

Permissions

The API user must belong to the same group as the specified ad campaign. Note that users with the Account Admin permission are members of all groups in an account, and users with the client Admin permission are members of all groups in all client accounts.

Example

The following JSON-RPC request and response describe a successful call to the **ad.calldna_postback.get** method. The response returns the Call DNA Ready URL, "*http://mydomain.com/calldna_postback*", for the ad campaign specified in the request.

Example Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.calldna_postback.get",
  "params": [
    "QrOxxxE9-fAxxxAD"
  ]
}
```

Example Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "http://mydomain.com/calldna_postback"
}
```

ad.calldna_postback.set

Sets the Call DNA Ready Postback URL for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.calldna_postback.set",
  "params": [
    "campaign_id",
    "end_point_URL"
  ]
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

campaign_id

Required string. The unique, system-generated ID that identifies an ad campaign.

end_point_URL

Required string, which can contain up to 256 characters. The endpoint URL Marchex should send data to for the specified ad campaign.

If the campaign already has a URL, it is replaced with the specified URL. You can specify an empty string (two double quotes) to delete a URL. Refer to [RFC1738: Uniform Resource Locators \(URL\)](#) for detailed information on the characters safe to use in URLs.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1
  "result": integer
}
```

Response parameters

result

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must be a Group Admin of the group that the specified campaign belongs to. Note that users with the Account Admin permission are Group Admins of all groups in an account, and parent account users with the Client Admin permission are Group Admins of all groups in all sub-accounts.

Example

The following JSON-RPC request and response describe a successful call to the `ad.calldna_postback.set` method. The request sets the call DNA ready postback end point URL for the specified ad campaign to `"http://mydomain.com/calldna_postback"`.

Example Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.calldna_postback.set",
  "params": [
    "QrOXxEE9-fATtgAD",
    "http://mydomain.com/calldna_postback"
  ]
}
```

Example Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
```



```
}
  "result": 17
}
```

ad.callmodel_postback.get

Gets the endpoint URL for the specified ad campaign.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.callmodel_postback.get",
  "params": [
    "campaign_id"
  ]
}
```

Request parameters

campaign_id

Required string. The unique, system-generated ID that identifies an ad campaign.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": "string"
}
```

Response parameters

result

A string containing the Call Model Postback URL for the specified ad campaign.

Permissions

The API user must belong to the same group as the specified ad campaign. Note that users with the Account Admin permission are members of all groups in an account, and users with the Client Admin permission are members of all groups in all client accounts.

Examples

Example Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.callmodel_postback.get",

```

```
"params": [  
  "QrOXxEE9-fATtgAD"  
]  
}
```

Example Response

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "result": "http://mydomain1.com"  
}
```

ad.callmodel_postback.set

Sets the endpoint URL for the specified ad campaign.

Request syntax

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "method": "ad.callmodel_postback.set",  
  "params": [  
    "campaign_id",  
    "end_point_URL"  
  ]  
}
```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

`campaign_id`

Required string. The unique, system-generated ID that identifies an ad campaign.

`end_point_URL`

Required string, , which can contain up to 256 characters. The endpoint URL Marchex should send data to for the specified ad campaign. If the campaign already has a URL, it is replaced with the specified URL. You can specify an empty string (two double quotes) to delete a URL. refer to RFC1738: Uniform Resource Locators (URL).

Response syntax

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "result": integer  
}
```

Response parameters

`result`

An integer that indicates successful completion of the method. If the method was unsuccessful, an error object is returned instead of this result value.

Permissions

The API user must be a Group Admin of the group that the specified campaign belongs to. Note that users with the Account Admin permission are Group Admins of all groups in an account, and parent account users with the Client Admin permission are Group Admins of all groups in all sub-accounts.

Example

The following JSON-RPC request and response describe a successful call to the `ad.callmodel_postback.set` method. The request sets the call model ready postback end point URL for the specified ad campaign to `http://mydomain.com/callmodel_postback`.

Example Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "ad.callmodel_postback.set",
  "params": [
    "CtjSZ1G1CNZF0AAm",
    "http://mydomain.com/callmodel_postback"
  ]
}
```

Example Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": 17
}
```

Rewrite Rules and JavaScript Rewrites

The following methods focus on the management of rewrite rules and JavaScript rewrites.

Method	Description
js.rewrite.list	Lists the JavaScript rewrite rules associated with the specified account.
js.rewrite.new	Adds a new JavaScript rewrite rule associated with the specified account.

[js.rewrite.list](#)

Lists the JavaScript [rewrites](#) associated with the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "js.rewrite.list",
  "params": [
    "account_id"
  ]
}
```

Request parameters

The request contains the following parameter:

account_id

Required string. The unique, system-generated account ID of the account associated with the JavaScript rewrites.

Response syntax

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "rwid": "string",
      "cmps": ["string1", "string2", ... "stringn"],
      "refdom": "string",
      "keyw": "string"
    },
    ...
  ]
}
```

Response parameters

The result is an array that contains 0 or more objects. Each object contains the following key-value pairs:

rwid

A string that contains the unique, system-generated rewrite ID of the JavaScript rewrite.

cmps

An array of strings. Each string contains the unique, system-generated campaign ID of an ad campaign that is associated with the JavaScript rewrite identified in *rwid*.

refdom

A string that contains the [referring domain](#) in the JavaScript rewrite. If the rewrite does not have a referring domain, an empty string (two double quotes) is returned.

keyw

A string (which can contain up to 255 characters) containing the [search phrase](#) in the JavaScript rewrite.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

The **js.rewrite.list** method returns rewrites associated with both [static](#) and [dynamic tracking](#) ad campaigns. For rewrites associated with a dynamic tracking campaign, the *keyw* parameter is always an empty string.

Example

The following JSON-RPC request and response describe a successful call to the **js.rewrite.list** method. The response returns an array of objects, each of which represents information about a JavaScript rewrite associated with the specified account.

Note: The response has been truncated for readability.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "js.rewrite.list",
  "params": [
    "CtjSZVRHBPk5igDz"
  ]
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": [
    {
      "rwid": "CmymM1OGF8IqxAEW",
      "cmps": ["CmymM1OGF74qxAET"],
      "refdom": "mydomain.com",
      "keyw": ""
    },
    {
      "rwid": "CmymM1Nn5f4qygDS",
      "cmps": ["CtjSZlG1CNZF0AAm", "CmymM1OGF74qxAET"],
      "refdom": "otherdomain.com",
      "keyw": "keyword"
    },
    ...
  ]
}
```

js.rewrite.new

Adds a new JavaScript [rewrite](#) to the specified campaigns in the specified account.

Request syntax

```
{
  "jsonrpc": "2.0",
```

```

    "id": 1,
    "method": "js.rewrite.new",
    "params": [
        "account_id",
        "referrer",
        "search_phrase",
        ["campaign_id1", "campaign_id2", ... "campaign_idn"]
    ]
}

```

Request parameters

The request contains the following positional parameters. All of the parameters are required, and they must be in the order listed in the Request syntax.

account_id

Required string. The unique, system-generated account ID that you want to create the rewrite rule in. You can use ["acct.list" on page 59](#) or ["acct.search" on page 78](#) to get the account ID.

referrer

Required string. The name of the referring domain. This can be the actual referrer (such as google.com, bing.com, or yahoo.search.com), or it can be a name used to identify a paid ad. Specifying a referring domain allows you to trigger the rewrite based on where—an organic search result, a paid ad, or a third-party website—a visitor clicks the link to your landing page. The text that you specify for the *referrer* must exactly match the `_vsrefdom` parameter in your landing page URL. See the ["Allowed Characters for the Referring Domain" on the next page](#) for additional details.

If you want the rewrite to trigger no matter where a visitor clicks a link to your landing page, set the referrer to an empty string (two double quotes). An empty string for the *referrer* corresponds to "any" in the Referrer column when you look at the JavaScript rewrite in the Call Analytics user interface.

search_phrase

Required string. This is the exact word you want visitors to have searched by to find and navigate to your website. To trigger a rewrite, there must be a URL parameter in your landing page URL that has a *value* that exactly matches what you specify here (including case).

If you want the rewrite to trigger no matter what word the visitor searched for, set the *search_phrase* to an empty string. An empty string for the *search_phrase* corresponds to "any" in the Search Phrases column when you look at the JavaScript rewrite in the Call Analytics user interface.

campaigns

Required array of strings. Each string contains the unique, system-generated campaign ID of a [static ad campaign](#) to be associated with the JavaScript rewrite. Although the request will accept an empty array (two square brackets), typically you associate at least one static ad campaign with the JavaScript rewrite.

Note Do not specify the campaign ID of a [dynamic tracking ad campaign](#). Although the request will appear to be successful, the actual results are undefined. Use ["ad.dynamic.new" on page 160](#) to create a rewrite for a dynamic tracking ad campaign. Although you can use ["ad.list" on page 169](#) or ["ad.list.all" on page 164](#) to get the campaign IDs, these methods may return campaign IDs for both dynamic tracking and static ad campaigns.

Response syntax

```

{
  "jsonrpc": "2.0",
  "id": 1,

```

```
{
  "result": "string"
}
```

Response parameters

The response consists of the following key-value pair:

result

A string that contains the unique, system-generated JavaScript rewrite ID of the new JavaScript rewrite.

Permissions

The API user must have Account Admin permissions in the specified account.

Remarks

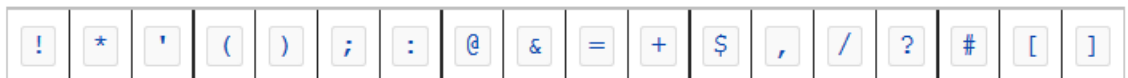
Allowed Characters for the Referring Domain

Some characters cannot be part of a URL (for example, spaces are not allowed), and some other characters have a special meaning in a URL and cannot be used in a URL parameter name or value. Because the value that you enter for the Referring Domain will be compared to the domain name in the [document.referrer](#) property and also to the `_vsrefdom` value (if present in the URL), you cannot use certain characters in the Referring Domain. (Note that the Call Analytics user interface does not prevent you from entering characters that are not allowed in a URL.)

The following characters are safe to use in the Referring Domain:

- » Upper case English letters: A to Z
- » Lower case English letters: a to z
- » Numbers: 0 to 9
- » Hyphen (or dash): -
- » Underscore: _
- » Period: .

CAUTION: Do not use spaces in the referring domain. Take care when you copy and paste from another application such as Microsoft Word because you may inadvertently copy characters that are not allowed in a URL. For example, what looks like a hyphen character may actually be an en dash or an em dash, or a beginning or trailing space may be added. Additionally, do not use any of the following characters:



Example

The following JSON-RPC request and response describe a successful call to the `js.rewrite.new` method. The request defines a new JavaScript rewrite, with both a referring domain and a search phrase, and associates it with the specified static ad campaign for the specified account.

Request

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "js.rewrite.new",
}
```

```
"params": [  
  "Qs0XxEE9-fATtgAE",  
  "otherdomain.net",  
  "chocolates",  
  ["CtjSZlGlCNzF0zAm"]  
]  
}
```

Response

```
{  
  "jsonrpc": "2.0",  
  "id": 1,  
  "result": "CnymMlOPoEcqxAEZ"  
}
```

**This page intentionally left blank to ensure new
chapters start on right (odd number) pages.**

CHAPTER 6

Glossary

This section defines terms used within the Call Analytics software and documentation.

account

A collection of users and ads in ad groups. A top-level or parent account can contain client accounts, which may also be referred to as child accounts or sub-accounts.

ad, ad campaign, or campaign

An item in an account that tracks call activity. Every call that comes in on a [call tracking number](#) is associated with an ad campaign.

ad group

A collection of ad campaigns and users. An ad campaign can be a member of only one ad group, but users can be members of many ad groups. Users who are members of an ad group can see calls to the group's ad campaigns in the Users who are members of an ad group can see calls to group's ad campaigns in the call log. Call Analytics can send an email to an ad group member for each good call, missed call, and voicemail to an ad campaign. You can filter reports by ad group. Also called *group*.

- » Every account has a normal ad group named `default`. If you don't need group features, just put your users and ads in this ad group.
- » Each ad group resides in just one account. Ad groups are not shared across accounts. The parent account has its own ad groups, as does each client account.

blacklist

A list of numbers that cannot call your clients' ads or just one client's ad. To use the blacklist, click the **Account** tab and click **Blacklist**.

breadcrumbs bar

The breadcrumbs bar indicates your current location on the web site. The breadcrumb links to the left are higher-level pages in the web site. You can jump to any level by clicking on the link.

Reports	Ad List	Call Log	Proxy Rewrites	JavaScript Rewrites
My Clients » ABC Plumbing » Call Tracking » Ad List				

Call DNA

Call DNA graphically displays conversations of phone calls to your ad campaigns in the call log in the Call Analytics user interface. Additionally, Call DNA analyzes the patterns in the conversation and classifies the result of the phone call. Whether or not you record phone calls, you can use the Call DNA visualization to see at a glance the outcome of the call.

call log

A page in the Call Analytics user interface that lists information about calls such as caller ID, ad campaign, time, duration, status, tags assigned, listened to, assigned to (nobody, or any user in the ad group), rating, and notes. Members of the ad campaign's ad group can click a call to view more information, see the Call DNA visualization of the call, or listen to the call if it was recorded.

call tracking number

A call tracking number (also referred to as an incoming number) is a phone number used to track calls to Call Analytics ad campaigns. A single call tracking number can be assigned to an ad campaign, or call tracking numbers can be dynamically assigned from a pool of numbers for use in dynamic tracking ad campaigns. You can obtain call tracking numbers in the Call Analytics User Interface on the Order Numbers pages.

campaign, ad, or ad campaign

An item in an account that tracks call activity.

carrier

A company that provides phone or data services. Carriers can operate wirelessly or over traditional wired land lines.

category

A named collection of client accounts. A client account can be in one category.

clean

The term "clean" means that a call tracking number is no longer in use by an ad campaign and that no calls have come in on the number in a month or more. See also: ["limbo" on page 357](#).

CSV file

A file format that contains tabular data (numbers and text) in plain-text form. You can use CSV files to import or export data in data analysis software programs. A CSV file can be viewed in a spreadsheet program such as Excel or Google Docs.

Document Object Model (DOM)

When you access a webpage, the browser creates a Document Object Model (DOM) of the page, which is referred to as a document. The DOM is an application programming interface (API) for HTML documents that defines the logical structure of a document and the way you can access and change the document.

document.referrer

The document interface represent a webpage loaded in a browser and serves as an entry point into the webpage's content. The document.referrer property contains the URL that was used to reach the current webpage and is generally called the [referrer](#). If the URL was typed directly into the browser's address field, this property will be empty.

Dynamic Tracking

Dynamic tracking, also referred to as Keyword Level Tracking (KWL T), is a Call Analytics feature that enables you to understand what keyword or session drove a specific call to your business or call center. With dynamic tracking, your campaigns use a pool of [call tracking numbers](#). At a high level, when a visitor reaches your website, a JavaScript snippet runs that replaces your business phone number with a phone number from the dynamic pool. The phone number is leased to the visitor for a set period of time. If the visitor calls the phone number within the lease period, Call Analytics can tie the call data back to the website visit.

E.164

E.164 is the standard format for international phone numbers, and it includes all the relevant information to route calls worldwide. E.164 numbers have a maximum of fifteen digits and are prefixed with a "+" followed by the country code and then the subscriber's phone number including area code.

EROM

An identifier in a report that specifies the exact report parameters in Call Analytics. Also called *erom_id*.

Extensions

A Call Analytics feature for [ad campaigns](#). An ad campaign with Extensions enabled plays a greeting and lets the caller enter an extension (or any numeric sequence, such as a zip code). Extensions allow you to create a simple [IVR](#), and have Call Analytics forward the call based on the caller's selection.

forward to phone number

The phone number that an ad campaign forwards calls to. Depending on the configuration of the ad campaign's forwarding rules, calls may be forwarded to more than one phone number.

ISO 8601

ISO 8601 is the international standard for the representation of dates and times. For example:

2016-11-29T14:06:15.286-08:00

- » **2016-01-29**: The date in YYYY-MM-DD format, which means: four digits for the Year, two digits for the Month, and two digits for the Day.
- » **T14:06:15.286**: The time portion starts with a **T** and may be in hh:mm:ss.sss format, which means:
 - » **hh**: Two digits for the hour (00 through 23). For example, 14:00 is 2 PM.
 - » **mm**: Two digits for the minute (00 through 59). For example: 14:06 is 2:06 PM.
 - » **ss**: Two digits for the second (00 through 59). For example: 14:06:15 is 15 seconds after 2:06 PM.
 - » **s**: One or more digits representing a fraction of a second. For example, 14:06:15.286 is 15 seconds and 286 milliseconds after 2:06 PM.

The granularity of the time portion may vary. For example, hh:mm:ss (without the milliseconds) and hh:mm (without the seconds) are both valid ISO 8601 formats.

- » **-08:00**: The time zone designator, which may be represented by an offset from [UTC](#).
- » If the time zone designator starts with a minus (-) sign, you subtract the specified number of hours from UTC. For example, -08:00 means 8 hours earlier than UTC, which is Pacific Standard Time. Note that the offset changes for Daylight Savings Time. For example, -07:00 is Pacific Daylight Time.
- » If the time zone designator starts with a plus (+) sign, you add the specified number of hours to UTC. For example, +01:00 means 1 hour later than UTC, which is Central European Time.
- » Zero offset from UTC is typically represented by the letter Z, for example: 2016-11-29T14:06:15.286Z.

IVR

Interactive voice response. An IVR allows callers to interact with a company's phone system via a phone keypad or by speech recognition, and it routes calls based on the caller's selection.

landing page URL

Also referred to as a destination URL, a final URL, or tracking URL. A landing page URL is the address of the webpage in your website that people reach when they click your ad. Typically, a landing page URL contains parameters that you are interested in tracking, and it is different from the display URL, which is the webpage address that is displayed with your ad.

limbo

The stage a call tracking number goes into when it is taken out of an active ad campaign. By default, a number remains in the limbo state for at least a month and is unavailable for use to ensure it is "clean" for the next ad campaign. Once removed from an ad campaign, a call tracking number stays in limbo until it has gone a month without a call. Once a month has passed without any calls, the number switches to the available status and becomes usable for a new ad campaign. You can manually change a number in limbo to available by Unlimboing the number from the Number List.

mkwid

A Marid ID, which is used in the integration with the Marin bid management system. The mkwid is a URL parameter that Marin users add to their [landing page URL](#) and is in the following format:

```
mkwid=[distribution][uniq_id]|pcrid|[creative_id]|pkw|[keyword_text]|pmt|[matchtype]|pdv|
[device]
```

North American Numbering Plan (NANP)

The North American Numbering Plan (NANP) is a phone numbering plan that includes twenty countries primarily in North America, but it also includes countries in the Caribbean and U.S. territories. Not all North American countries participate in the NANP. Specifically, Mexico and countries in Central America do not participate. A NANP phone number is 10 digits long and consists of a three-digit area code followed by a seven-digit local phone number (for example: "8005551212"). Often you see NANP phone numbers with dashes after the area code or parentheses around the area code (for example: "800-555-1212" or "(800)555-1212").

NPA and NXX

Number Plan Area (area code) and three-digit exchange that make up the first 6 digits of a 10-digit telephone number.

playfile

Playfiles play a greeting to callers to an ad campaign. Playfiles customize the caller experience and provide callers with details about the business or service they have reached.

postback

A one-time, one-way internet connection that notifies a server of an event, such as a call starting, a call ending, or submission of a web-based form.

referrer

The website where a visitor clicked a link to reach your landing page. The referrer to your landing page can be found in the HTML "[Document Object Model \(DOM\)](#)" [on page 355](#) referrer property. When configuring a rewrite in the Call Analytics user interface, you have the option of configuring a Referring Domain. This can be the actual referrer (such as google.com, bing.com, or yahoo.search.com), or it can be a name used to identify a paid ad.

rewrite

Rewrites change a website to show your ad campaign's call tracking number (or a tracking number from a dynamic number pool) based on how the visitor reached the website. A rewrite can replace a phone number on a website for every visitor, or only for visitors from a particular referring website or a particular paid ad.

rewrite rule

A Rewrite contains one or more rewrite rules that specify which phone numbers on a website should be replaced with a call tracking number and which phone numbers should be excluded from the rewrite.

search phrase

An optional value in a JavaScript rewrite that specifies the keyword (for example, "cars" or "hotels") that a visitor to your landing page entered in a search engine that will trigger the rewrite.

search campaign

Refers to an online advertising campaign set up in a search engine, such as Google AdWords or Bing.

static campaign

Refers to a Call Analytics ad campaign that is configured with a single [call tracking number](#). This type of campaign is also referred to as a one-to-one campaign because it has one call tracking number, in contrast to [dynamic tracking campaigns](#).

step

A step is an event in an ad campaign call-forwarding sequence. If a forward-to number at step 1 does not answer before the timeout period ends, Call Analytics rings the forward-to number at step 2. Same-value steps can also cause simultaneous ringing.

tag

In Call Analytics, a tag is an identifier in an ad campaign that can be added to a call. A call can include one tag.

In third-party tag management systems such as Google Tag Manager, a tag is a snippet of code that runs on your website and sends information to a third party, such as Call Analytics or Google Analytics.

user

A member of an account. A user is identified in the Call Analytics platform by their email address. Users can be assigned permissions that govern what functionality they access to in the Call Analytics UI and API. Users created in a parent account can be granted access to client accounts.

UTC

UTC (Coordinated Universal Time in English), is the international standard for regulating clocks and time, which is based on the 0° longitude meridian, also known as the Greenwich meridian. The various time zones around the world are indicated via an offset from UTC. For example:

- » -05:00 means 5 hours earlier than UTC, which is Eastern Standard Time.
- » +01:00 means 1 hour later than UTC, which is Central European Time.

The switch to daylight saving time does not affect UTC itself, but does affect the offset. For example, Eastern Standard Time is -05:00 and Eastern Daylight Time is -04:00.

Voicemail

A Call Analytics feature for [ad campaigns](#). An ad campaign with Voicemail enabled plays a greeting and lets the caller record a message. Call Analytics then forwards the message to the email address configured on the ad campaign.

weight

Specifies call distribution at a specific step. Two forward-to numbers at the same step with the same weight receive equal numbers of incoming calls.

whitelist

A list of numbers that will not be blocked from reaching ad campaigns, even if their calling patterns are unusual, such as calling a campaign before it is live. Numbers that will be used to test the campaign should be added to the whitelist.

Wind Down

Wind Down is an optional period that you can set when you deactivate an ad campaign. During the Wind Down period, near the end of an ad campaign, a caller hears a Wind Down playfile or a new number to call in the future.