



# API Technical Guide: Search

Cheetah Messaging

# Table of Contents

<b>1 Introduction</b>	<b>4</b>
Purpose	4
Overview	4
Methods	4
Authentication	5
<b>2 Search for Items</b>	<b>6</b>
Overview	6
Retrieve Item Types	6
count / page	6
Retrieve an Item by Object ID	7
id	7
Retrieve an Item by Object Reference ID	7
refId	8
type	8
Retrieve Items by Name	8
name	8
count / page	8
Retrieve Items by Type	9
type	9
count / page	9
Retrieve Items in a Folder	10
folderId	10
type	10
name	10
count / page	11
<b>3 Response</b>	<b>12</b>
Success	12
Errors	13
<b>4 Sample Messages</b>	<b>14</b>
Sample Response	14
<b>5 Appendix A -- Identifiers</b>	<b>15</b>
Object ID	15
Object Reference ID	16
Folder ID	18
<b>6 Appendix B -- Item Types</b>	<b>19</b>



# 1 Introduction



## Purpose

The purpose of this document is to provide an overview of the **SEARCH** API endpoint within the Cheetah Messaging platform. This document discusses the intended use of the **SEARCH** endpoint, and provides technical details for how to implement the endpoint.

## Overview

The **SEARCH** endpoint (also referred to as the "Object" endpoint) allows you to search for, and retrieve information about specified items, or assets, within the platform. The different methods provide several different ways of searching for the desired item, such as by name, by type, or by folder location, for example. This endpoint is very useful for looking up Object ID and Object Reference ID values for items, so you can reference those items in other API endpoints.

This endpoint requires authentication using OAuth 2.0, and supports JSON and XML messages.

The URLs for this endpoint are:

- **North America:** <https://api.eccmp.com/services2/api/Object>
- **Europe:** <https://api.ccmp.eu/services2/api/Object>
- **Japan:** <https://api.marketingsuite.jp/services2/api/Object>

## Methods

The **SEARCH** endpoint supports the following HTTP methods:



- **GET:** Retrieve a list of all item types
- **GET:** Retrieve an item by Object ID
- **GET:** Retrieve an item by Object Reference ID
- **GET:** Retrieve items by Name
- **GET:** Retrieve items by Item Type
- **GET:** Retrieve all items in a folder -- by Item Type and Name

## Authentication

Access to the **SEARCH** endpoint requires that you first be authenticated within the platform. Within Messaging, authentication is handled by OAuth 2.0. To authenticate with OAuth 2.0, you must first obtain a "Consumer Key" and a "Consumer Secret." Both of these values are managed at the user level, and can be obtained from within the Messaging application.

Next, you'll use your Consumer Key and Consumer Secret to request a "token." A token is a text string that, when provided in a request message, will allow the user access to the requested service. Tokens are valid only for a certain period of time.

For more details on how to authenticate your API request, please see the *Messaging: API How-to Guide*.



# 2 Search for Items

## Overview

This section describes how to search for items using the **SEARCH** endpoint.

## Retrieve Item Types

Many of the methods supported by the **SEARCH** endpoint require you to provide the desired "Item Type," such as "Filter" or "SeedList" for example. The "Retrieve Item Types" option simply returns a list of all valid Item Types. For a list of these values, please see [Appendix B](#).

The parameters when retrieving all Item Types are described below.

### count / page

These integer parameters are optional.

These two parameters are used to control how many Item Types, and which Item Types, should be included within the response message. If you don't provide a value for these parameters, the system will default both to blank, meaning that the system will return all Item Types. However, you can use the **count** and **page** parameters to alter the response message to include a different quantity and selection of Item Types.

The **count** parameter will split the response message up into "pages" of the indicated size. The **page** parameter then tells the system which page you want to see in the response message.

### Note

The **page** parameter start with "1" as the first page, "2," for the second page, and so forth.



If used, these parameters must be sent as query type parameters within the URL. For example:

```
https://api.eccmp.com/services2/api/Object?page=1&count=10
```

## Retrieve an Item by Object ID

This version of the **SEARCH** endpoint allows you to retrieve the details of a single Item by providing its **Object ID**.

The parameters when retrieving an Item by Object ID are described below.

### id

This integer parameter is required.

The **id** parameter represents the **Object ID** of the desired Item. The **id** must be sent as a query type parameter within the URL. For example:

```
https://api.eccmp.com/services2/api/Object?id=38129
```

## Retrieve an Item by Object Reference ID

This version of the **SEARCH** endpoint allows you to retrieve the details of a single Item by providing its **Object Reference ID**. The Object Reference ID is a system-generated identifier that's unique to every item, within an Item Type, in your account.

When using this version of the **SEARCH** endpoint, you must provide both the Object Reference ID and the Item Type. The Object Reference ID is not unique across your entire account; it's unique only within an Item Type. For example, you could have a Filter, a Campaign, and a Content Block with the same Object Reference ID value. Conversely, those three assets would all have different Object IDs. Therefore, in order to search for a specific item by Object Reference ID, you need both the Object Reference ID AND the Item Type.

The parameters when retrieving an Item by Object Reference ID are described below.



## refId

This integer parameter is required.

The **refId** parameter represents the **Object Reference ID** of the desired Item. The **refId** must be sent as a query type parameter within the URL. For example:

```
https://api.eccmp.com/services2/api/Object?refId=37681&type=Filter
```

## type

This string parameter is required.

The **type** parameter represents the **Item Type** for the asset for which you're searching. The **type** parameter must be sent as a query type parameter within the URL. For example:

```
https://api.eccmp.com/services2/api/Object?refId=37681&type=Filter
```

# Retrieve Items by Name

This version of the **SEARCH** endpoint allows you to retrieve the details of one or more Items (across all Item Types) by providing all or part of the Item's name.

The parameters when retrieving Items by name are described below.

## name

This string parameter is required.

The **name** parameter represents all, or part of, the display name of the desired Item. The **name** must be sent as a query type parameter within the URL. For example:

```
https://api.eccmp.com/services2/api/Object?name=test
```

## count / page

These integer parameters are optional.

These two parameters are used to control how many Items, and which Items, should be included within the response message. If you don't provide a value for these parameters, the system will default both to blank, meaning that the system will return all Items that match the provided search criteria. However, you can use the **count** and **page** parameters to alter the response message to include a different quantity and selection of Items.



The **count** parameter will split the response message up into "pages" of the indicated size. The **page** parameter then tells the system which page you want to see in the response message.

### Note

The **page** parameter start with "1" as the first page, "2," for the second page, and so forth.

If used, these parameters must be sent as query type parameters within the URL. For example:

```
https://api.eccmp.com/services2/api/Object?name=test&page=1&count=10
```

## Retrieve Items by Type

This version of the **SEARCH** endpoint allows you to retrieve the details of one or more Items of a specified Item Type.

The parameters when retrieving Items by Type are described below.

### type

This string parameter is required.

The **type** parameter represents the **Item Type** for the asset for which you're searching. The **type** parameter must be sent as a query type parameter within the URL. For example:

```
https://api.eccmp.com/services2/api/Object?type=Filter
```

### count / page

These integer parameters are optional.

These two parameters are used to control how many Items, and which Items, should be included within the response message. If you don't provide a value for these parameters, the system will default both to blank, meaning that the system will return all Items that match the provided search criteria. However, you can use the **count** and **page** parameters to alter the response message to include a different quantity and selection of Items.



The **count** parameter will split the response message up into "pages" of the indicated size. The **page** parameter then tells the system which page you want to see in the response message.

### Note

The **page** parameter start with "1" as the first page, "2," for the second page, and so forth.

If used, these parameters must be sent as query type parameters within the URL. For example:

```
https://api.eccmp.com/services2/api/Object?type=Filter&page=1&count=10
```

## Retrieve Items in a Folder

This version of the **SEARCH** endpoint allows you to retrieve the contents of a specified Folder, narrowed down by Item Type, and also optionally by name.

The parameters when retrieving Items in a Folder are described below.

### folderId

This integer parameter is required.

The **folderId** represents the **Folder ID** of the Folder in which you want to search. The **folderId** parameter must be sent as a query type parameter within the URL. For example:

```
https://api.eccmp.com/services2/api/Object?folderId=37249&type=Filter
```

### type

This string parameter is required.

The **type** parameter represents the **Item Type** for the asset for which you're searching. The **type** parameter must be sent as a query type parameter within the URL. For example:

```
https://api.eccmp.com/services2/api/Object?folderId=37249&type=Filter
```

### name

This string parameter is optional.



The **name** parameter represents all, or part of, the display name of the desired Item. The **name** parameter is used in conjunction with the **type** parameter described above. These two parameters are joined by an AND operator, meaning the platform will return only the assets that meet both criteria.

If used, the **name** parameter must be sent as a query type parameter within the URL. For example:

```
https://api.eccmp.com/services2/api/Object?folderId=37249&type=Filter&name=audience
```

## count / page

These integer parameters are optional.

These two parameters are used to control how many Items, and which Items, should be included within the response message. If you don't provide a value for these parameters, the system will default both to blank, meaning that the system will return all Items that match the provided search criteria. However, you can use the **count** and **page** parameters to alter the response message to include a different quantity and selection of Items.

The **count** parameter will split the response message up into "pages" of the indicated size. The **page** parameter then tells the system which page you want to see in the response message.

### Note

The **page** parameter start with "1" as the first page, "2," for the second page, and so forth.

If used, these parameters must be sent as query type parameters within the URL. For example:

```
https://api.eccmp.com/services2/api/Object?folderId=37249&type=Filter&name=audience&page=1&count=10
```



# 3 Response

This section describes the possible response messages sent back from the **SEARCH** endpoint.



## Success

A successful response to the **Retrieve Item Types** version of the **SEARCH** endpoint will generate a response code of "200," followed by a list of all valid Item Type values (see **Appendix B** for this list of valid values).

A successful response to all other versions of the **SEARCH** endpoint will generate a response code of "200," followed by the details of the Item, or Items, that met the provided search criteria. The response will include the following information for each Item:

- **obj\_id** -- The Object ID
- **display\_name** -- The name of the item
- **type\_id** -- The Item Type
- **ref\_id** -- The Object Reference ID
- **parent\_obj\_id** -- The Folder ID for the Folder where the Item is located
- **eligibility\_status\_id** -- The status of the Item -- either "DRAFT" or "READY." Items in DRAFT status can't be used in Campaigns.

A "null" response message means your request parameters were valid, but no items were found matching the provided search criteria.



## Errors

If Messaging encounters a problem with a **SEARCH** request message, the platform will send an "error" message with details of the problem. Below is a list of error codes and their descriptions.

Response Code	Error message	Description
400	"Invalid Object Type"	Invalid Item Type; see <a href="#">Retrieve Items Types</a> for details on how to get a list of valid Item Types. <a href="#">Appendix B</a> also provides a list of valid values.
400	"User does not have access to this resource"	Invalid Object ID, or you don't have access to the provided Object ID



# 4 Sample Messages

## Sample Response

For all of the search methods available through the **SEARCH** endpoint, the response message provides the same basic information for the asset, or assets, that met the provided search criteria.

Below is a sample response message.



```
{
  "obj_id": 55527,
  "display_name": "Seed List Test",
  "type_id": "SeedList",
  "ref_id": 47092,
  "parent_obj_id": 11440,
  "eligibility_status_id": "READY"
},
{
  "obj_id": 55055,
  "display_name": "Internal seeds",
  "type_id": "SeedList",
  "ref_id": 46657,
  "parent_obj_id": 11440,
  "eligibility_status_id": "READY"
},
{
  "obj_id": 54437,
  "display_name": "Decoy Seed List",
  "type_id": "SeedList",
  "ref_id": 2919,
  "parent_obj_id": 11440,
  "eligibility_status_id": "READY"
}
```



# 5 Appendix A -- Identifiers

Messaging uses several different types of IDs when referencing assets, such as tables, fields, folders, Filters, and so forth. This appendix describes these different types of IDs, and provides steps on how to look up the value of an ID.

## Object ID

The Object ID is a system-generated identifier for every asset in your account.

For many item types, the value for this identifier can be found within the Messaging application by navigating to the details screen for that item.

To find the Object ID within the application:

1. From the System Tray, select the screen for the desired Item Type. The system displays a list of all the items in your account.
2. Select the desired item. The item's detail screen is displayed.
3. In the Tool Ribbon, click the first tab; the name of this tab is usually the name of the item type, like "Filter" for example. The "Item Details" screen is displayed. The Object ID is listed on this screen.



FILTER
EDIT

Item Details

Related Items

### Item Details & Revision History

*which users created/modified this item and its system ids*

Modified	11/14/2019 12:55 PM [ Thomas Anderson ]
Created	8/11/2017 10:19 AM [ Thomas Anderson ]
Owner	Thomas Anderson [ <a href="#">change</a> ]
Obj Id	46435
Obj Ref Id	37681

You can also look up the Object ID using the **SEARCH** endpoint, such as [Retrieve Items by Name](#), or [Retrieve Items by Type](#), for example. As part of the API response message, the system provides the Object ID in the **obj\_id** parameter. For example:

```

{
  "obj_id": 46435,
  "display_name": "Summer campaign audience",
  "type_id": "Filter",
  "ref_id": 40329,
  "parent_obj_id": 37249,
  "eligibility_status_id": "READY"
}

```

## Object Reference ID

The Object Reference ID is a system-generated identifier for every asset within an Item Type, in your account.

For many item types, the value for this identifier can be found within the Messaging application by navigating to the details screen for that item.



To find the Object Reference ID within the application:

1. From the System Tray, select the screen for the desired Item Type. The system displays a list of all the items in your account.
2. Select the desired item. The item's detail screen is displayed.
3. In the Tool Ribbon, click the first tab; the name of this tab is usually the name of the item type, like "Filter" for example. The "Item Details" screen is displayed. The Object Reference ID is listed on this screen.

Field	Value	User
Modified	11/14/2019 12:55 PM	[ Thomas Anderson ]
Created	8/11/2017 10:19 AM	[ Thomas Anderson ]
Owner	Thomas Anderson	[ change ]
Obj Id	46435	
Obj Ref Id	37681	

You can also look up the Object Reference ID using the [SEARCH](#) endpoint, such as [Retrieve Items by Name](#), or [Retrieve Items by Type](#), for example. As part of the API response message, the system provides the Object Reference ID in the `ref_id` parameter. For example:

```
{
  "obj_id": 46435,
  "display_name": "Summer campaign audience",
  "type_id": "Filter",
  "ref_id": 40329,
  "parent_obj_id": 37249,
  "eligibility_status_id": "READY"
```



```
}
```

## Folder ID

The Folder ID is a unique, system-generated identifier for each folder and sub-folder in your system. This value is not displayed within the application user interface anywhere, so to get your Folder ID, you must retrieve it by means of the [SEARCH](#) API endpoint. The easiest method is to use the version that lets you [Retrieve Items by Type](#) -- use a **type** value of "Folder." For example:

```
https://api.eccmp.com/services2/api/Object?type=Folder
```

The response message provides a list of all the folders in your system. Find the desired folder in the response message. As part of the API response message, the system provides the Folder ID, which is referred to as the **obj\_id**.

```
{  
  "obj_id": 37249,  
  "display_name": "Promo Campaigns",  
  "type_id": "Folder",  
  "ref_id": 37249,  
  "parent_obj_id": 11440,  
  "eligibility_status_id": "READY"  
}
```

### Note

If this Folder is a sub-folder, the **parent\_obj\_id** is the Folder ID of the parent folder.



# 6 Appendix B -- Item Types

Below is a list of all the valid Item Type values.

Item Type	Description
AlertList	Alert Groups (Email, SMS, and Push channels)
AlertListEmail	Alert Groups (Email channel only)
AlertListSms	Alert Groups (SMS Text channel only)
AnalyticsVendorConnectorGroup	Analytic Vendor Connectors
ApiCampaign	
ApiRetrieve	
Attachement (please note spelling)	Attachments
AuthenticationSet	Web Authentication
BncCat	
Campaign	Campaigns (all channels)
CampaignControl	Campaigns (Control channel only)
CampaignDataFile	Campaigns (Data File and Print channels only)
CampaignEmail	Campaigns (Email channel only)
CampaignHttp	Campaigns (Webhook channel only)
CampaignSms	Campaigns (SMS Text channel only)
CampaignSocialFacebook	
CampaignSocialFacebookAd **	Facebook External Ads
CampaignSocialFacebookAnon	Facebook Wall Posts
CampaignSocialLine	Campaigns (LINE channel only)
CampaignSocialTwitter **	Campaigns (Twitter channel only)
CampaignSocialTwitterAnon **	Tweets
CampaignStub	Campaigns (Tracking Code channel)
CampaignWeb	Web Pages
CampaignWebXml	Campaigns (XML / RSS channel only) **
CampCell	Individual Cells within a Split Cells Campaign



Item Type	Description
CampMetaParamOption	
Content	
ContentBlock	Content Blocks
ContSource	
CustApi	
CustForwardProfile	Forwarding Handlers
CustHome	
CustomChannelType	
Customer	
CustReplyProfile	Auto Reply Handlers
DataFileLayout	Data File Templates
DbDiagram	Database Diagrams
DeliverabilityList	
DeliverabilityListEmail	
DynamicBlock	Dynamic Blocks
DynamicBlockData	Dynamic Data Files
Entities	
Entity	
ExclusionList	Exclusion Lists
Export	Exports
FacebookAccount **	
FacebookAdAccountSetup **	
FacebookAdAudienceExport **	Facebook Audience Exports
FacebookApplication **	
FacebookData **	
FacebookPage **	
Filter	Filters
Folder	Folders (for Campaigns, assets, etc.)
FromAddress	



Item Type	Description
FtpProfile	FTP Profiles
FtpTemplate	FTP Import Templates
GoogleApplication	
HierarchyRule	Hierarchy Rules
HostedItem	
Import	Imports
ImportForm	Web Forms
ImportFormApi	API Post
ImportFormSfdc **	Salesforce Connector
JasperDashboard	
JasperDashboardDesigner	
JasperDesigner	
JasperReport	
JasperReportDesigner	
LineAccountSetup	
Link	
LinkGroup	
LinkLibrary	
LiveImage **	Live Images
LoopingBlock	Looping Blocks
MyFiles	
OmniureAnalyticsConnector	
PageflexProject	Print Templates
ParagraphFblike	Facebook Like Buttons
ParagraphGoogleplus **	Google Plus Buttons
ParagraphSocial	Share to Social Blocks
ParagraphUnsub	Opt-Out Messages
PlatformEventSubscription	Subscriptions
PostalProcessNcoa	NCOA Processes



Item Type	Description
PostalProcessPresort	Postal Pre-sorts
PrintPiece	Print Pieces
Program	Programs
ProgramCell	
ProgramDomain	PURL Domains
ProofList	Proofing Groups (all channels)
ProofListEmail	Proofing Groups (Email channel only)
ProofListSms	Proofing Groups (SMS Text channel only)
Property	
PropertyMap	Data Maps
Push_Campaign	Campaigns (Push Notification channel only)
Push_Registration	Device Registrations
RecordLookup	
RecycleBin	
ReportCampaign	Campaign Reports
ReportCompare **	Comparison Reports
ReportContent **	Content Reports
ReportCustom **	Custom Reports
ReportDatabase	
ReportForm **	Form Reports
ReportGoal	Report Goal Sets
ReportSettings	
ResponseAttrResults	
ResponseAttrSetup	
SalesforceAccount **	Salesforce Accounts
Search	
SecurityPolicy	
SeedList	SeedLists
SenderProfile	Sender Profiles



Item Type	Description
SentimentRegex	
Shortcut	
SmsKeyword	
SmsKeywordGroup	SMS Keyword Groups
SmsSender	
SocialKeyword	
SocialKeywordGroup	Social Keyword Groups
SubCustomers	
SubscriptionList	
Template	Templates
TemplateDynamicBlock	
TemplateParagraph	Template Blocks
TwitterAccount **	
TwitterApplication **	
UnsubCat	
User	
UserGroup	
ViewCustom	
ViewExport	
ViewResponse	
ViewTable	
WebEvent	Web Events
WebFeed **	Web Feeds
WebFiles	
WorkflowStep	
WorkflowTemplate	Approval Workflows

\*\* These Item Types have been deprecated.

