



---

# **Communote REST API Plugin**

## **Version 3.2-SNAPSHOT**

### **Documentation**

---



## Table of Contents

1	<b>Table of Contents</b>	<b>i</b>
2	<b>Introduction</b>	<b>1</b>
3	<b>Getting Started</b>	<b>2</b>
4	<b>Usage of Resources</b>	<b>11</b>
5	<b>Available Resources</b>	<b>13</b>
6	<b>Attachment Resource and Subresources</b>	<b>15</b>
7	<b>Note Resource and Subresources</b>	<b>18</b>
8	<b>Tag Resource and Subresources</b>	<b>26</b>
9	<b>TagStoreTagList Resource and Subresources</b>	<b>29</b>
10	<b>TagSuggestionList Resource and Subresources</b>	<b>31</b>
11	<b>TimelineNote Resource and Subresources</b>	<b>38</b>
12	<b>TimelineTag Resource and Subresources</b>	<b>52</b>
13	<b>TimelineTopic Resource and Subresources</b>	<b>59</b>
14	<b>TimelineUser Resource and Subresources</b>	<b>66</b>
15	<b>Topic Resource and Subresources</b>	<b>73</b>
16	<b>User Resource and Subresources</b>	<b>100</b>
17	<b>Usage of Communote Client for Java</b>	<b>114</b>
18	<b>Imprint</b>	<b>134</b>



# 1 Introduction

---

This documentation describes in detail the Communote REST API Version 3.0. The Communote REST API Version 3.0 is supported by Communote 3.0

The resources of this API can be accessed directly by HTTP or HTTPS depending on the configuration of the Communote Installation to be used.

There exists also a Java Client that eases the access to the Communote REST API. This Java based `CommunoteClient` is provided on request.

In the first Chapter [Getting Started](#) the usage of the API is demonstrated by simple examples.

The chapter [Usage of Resources](#) describes in detail the resources that can be used to access Communote or perform certain actions.

Furthermore you find chapters for every resource provides by the Communote REST API Version 3.0. The chapter [Available Resources](#) gives you an overview of all resources.

The last chapter [Usage of CommunoteClient](#) gives you informations about important classes and available methods of `CommunoteClient` for Java.

## 2 Getting Started

In this chapter you find a simple example for [accessing resources direct by HTTP\(S\)](#) and by [using the Communate Client for Java](#).

### 2.1 Getting started with direct access by HTTP(S)

Here is a full example to create, get, edit and delete a topic and to list topic resources.

*Please read [Usage of Resources](#) for more information (e.g. attributes of responses). Also use the [Topic Resource](#) documentation to find more information about the topic resource.*

#### 2.1.1 Determining the API URL

The API URL varies depends on the usage of an Inhouse or the Public Installation with an Enterprise Account. The base path of the API for inhouse installation is by default: `http://example.com/microblog/global/api/rest/2.4` Here replace example.com with the host Communate is running on.

For the SaaS installations running on `http://www.communote.com/` the default base path has to be adapt by replacing `clientId` with the id of the Communate Enterprise Account to access: `http://www.communote.com/microblog/clientId/api/rest/2.4`

For secure access use HTTPS instead of HTTP.

#### 2.1.2 Authentication

At the moment only Basic Access Authentication is allowed. Therefore you should always use and force HTTPS communication for production services.

On the first request a cookie named `JSESSIONID` will be returned. The value of the session cookie `JSESSIONID` is the identifier of the session. If possible the session identifier should always be included in further request. Once a authentication has been successful within the session no further authentication is necessary. Hence the username and password needs only to be send one time, in the most use cases on the first request.

`Set-Cookie: JSESSIONID=70C45F4845D861EFC8E44DB066C1DC09; Path=`

In case it is not possible to send cookies within a request the session identifier can be added to the request URL. The session identifier must be included directly before the '?' of the url or at the end if no query string is included in the URL:

`http://example.com/microblog/global/api/rest/2.4/notes;jsessionid=70C45F4845D861EFC8E44DB066C1DC09?param=12`

Using the API without authentication will pop-up a Basic Authentication dialog in the browser. This can be avoided using the request parameter `noAuthenticationChallenge`:

`http://example.com/microblog/global/api/rest/2.4/notes?param=12&noAuthenticationChallenge=true`

This only is useful if there is some other authentication provided within the request (e.g. by cookie or sending the basic authentication header directly).

#### 2.1.3 Server Information

Providing the supported REST API versions plus information about the current user.

#### Request

<b>Request-Url</b>	http://example.com/microblog/global/api/rest/information
<b>Http-Method</b>	GET
<b>Content-Type</b>	application/json;charset=UTF-8

## Response

In the response you get information containing the supported REST-API versions. Furthermore, you can get information about the user when he is logged in.

<b>Status Code</b>	200 OK
<b>Content-Type</b>	application/json;charset=UTF-8

```
{
  "preferredVersion": "2.4",
  "userFirstName": "John",
  "userId": 1,
  "userLastName": "Doe",
  "userAlias": "jdo",
  "supportedVersions": ["2.4", "2.4", "1.3"],
  "license" : {
    "type": "OS", // one of "OS" (Online Service) or "ST" (Standalone)
    "isStandard": true, // If set, you are using the free standard variant
    "isTest": true, // If set, you are within the test phase of Communote
    "isValid": true, // If true, the current license is valid and it is pos
    "reason": "TOO_MANY_USERS" // A reason about the current license status
                                // This is one of "OKAY", "TOO_MANY_U
                                // or "VERSION_NOT_SUPPORTED". Log in
  }
}
```

### 2.1.4 Create Topic

This is an example to create a new topic by using the [Topic Resource](#).

#### Request

Creates a topic resource with title "example" and tag "new".

*More information about attributes and parameter of create (POST) topic under [POST Topic Resource](#).*

<b>Request-Url</b>	http://example.com/microblog/global/api/rest/2.4/topics/
<b>Http-Method</b>	POST
<b>Content-Type</b>	application/json;charset=UTF-8

```
{
  "title": "example",
  "tags": [ { "defaultName": "tag" } ]
}
```

### Response

As response you will receive a human readable message and a status of the request and the actual result of your request. In this example the result will be the identifier of the just created topic.

<b>Status Code</b>	200 OK
<b>Content-Type</b>	application/json;charset=UTF-8

```
{
  "message": "The topic was successfully created.",
  "status": "OK",
  "result": 1
}
```

#### 2.1.5 Get Topic

This example shows how to receive information about a topic using the topic resource.

### Request

Get the topic resource with identifier "1".

*More information about attributes and parameter of get topic under [GET Topic Resource](#).*

<b>Request-Url</b>	http://example.com/microblog/global/api/rest/2.4/topics/1
<b>Http-Method</b>	GET
<b>Content-Type</b>	application/json;charset=UTF-8

### Response

In the response you get a message, status of request and the identifier of topic in result.

<b>Status Code</b>	200 OK
<b>Content-Type</b>	application/json;charset=UTF-8



```
{
  "status": "OK",
  "result": {
    "topicId": "1",
    "lastModificationDate": 1331569395509,
    "topicEmail": null,
    "createSystemNotes": false,
    "description": null,
    "tags": [
      {
        "tagId": 1,
        "tagStoreAlias": "DefaultBlogTagStore",
        "tagStoreTagId": "new",
        "tagStoreType": null,
        "languageCode": "en",
        "name": "new",
        "description": null,
        "defaultName": "new"
      }
    ],
    "alias": "example",
    "title": "example",
    "userRole": "MANAGER",
    "isFollow": false,
    "properties": null,
    "allCanWrite": false,
    "allCanRead": false
  }
}
```

### 2.1.6 Edit Topic

This is an example to edit a topic.

#### Request

Edit a topic resource to set the new title "new example" and assign a tag with the name "new tag".

The request to edit a topic is tunneled over POST with the appendix parameter `_method=PUT` on url. Please read [Tunneling](#) for more information.

*More information about attributes and parameter of edit (PUT) topic under [LIST Topic Resource](#).*

<b>Request-Url</b>	<code>http://example.com/microblog/global/api/rest/2.4/topics/1?_method=PUT</code>
<b>Http-Method</b>	POST
<b>Content-Type</b>	<code>application/json; charset=UTF-8</code>

```
{
  "title": "new example",
  "tags": [ { "defaultName": "new tag" } ]
}
```

### Response

In the response you get a message, status of request and the identifier of topic in result.

<b>Status Code</b>	200 OK
<b>Content-Type</b>	application/json;charset=UTF-8

```
{
  "message": "The topic details have been updated
              successfully",
  "status": "OK",
  "result": 1
}
```

### 2.1.7 List Topics

This is an example to list topics.

#### Request

List topics where the user is MANAGER. The topicListType parameter is mandatory.

*More information about attributes and parameter of list topic under [LIST Topic Resource](#).*

<b>Request-Url</b>	http://example.com/microblog/global/api/rest/2.4/topics?topicListType=MANAGER
<b>Http-Method</b>	GET
<b>Content-Type</b>	application/json;charset=UTF-8

### Response

In the response you get a metaData element containing the number of all found topics and the information if there are more topics that can be used for for paging. Furthermore, you find the status of the request and the topic resource in result.

<b>Status Code</b>	200 OK
<b>Content-Type</b>	application/json;charset=UTF-8

```

{
  "metaData": {
    "numberOfElements": 100,
    "moreElementsAvailable": true
  },
  "status": "OK",
  "result": [
    {
      "topicId": "1",
      "lastModificationDate": 1331569395509,
      "topicEmail": null,
      "createSystemNotes": false,
      "description": null,
      "tags": [
        {
          "tagId": 1,
          "tagStoreAlias": "DefaultBlogTagStore",
          "tagStoreTagId": "new tag",
          "tagStoreType": null,
          "languageCode": "en",
          "name": "tag",
          "description": null,
          "defaultName": "tag"
        }
      ],
      "alias": "example",
      "title": "example",
      "userRole": "MANAGER",
      "isFollow": false,
      "properties": null,
      "allCanWrite": false,
      "allCanRead": false
    },
    ...
  ]
}

```

### 2.1.8 Delete Topic

This is an example to delete a topic.

#### Request

Delete the topic resource with identifier "1".

The request to edit a topic is tunneled over POST with the appendix parameter `_method=DELETE` on url.

Please read [Tunneling](#) for more information.

*More information about attributes and parameter of delete topic under [DELETE Topic Resource](#).*

<b>Request-Url</b>	<code>http://example.com/microblog/global/api/rest/2.4/topics/1?_method=DELETE</code>
<b>Http-Method</b>	POST

Content-Type	application/json;charset=UTF-8
--------------	--------------------------------

### Response

In the response you get a message and status of request.

Status Code	200 OK
Content-Type	application/json;charset=UTF-8

```
{
    "message": "The topic was deleted successfully.",
    "status": "OK"
}
```

## 2.2 Getting started with Communote Client for Java

This is the example for topic resource to getting started with the Communote Client for Java. For now we provide the Communote Client for Java on request. Please feel to contact us. You can find the full example in client sources in class TopicExample.java.

```
com.communote.plugins.api.rest.client.example.TopicExample.java
```

### Dependency

```
com.communote.plugins.api.rest.client.CommunoteClient;
```

### Source Code

To get an instance of CommunoteClient you must set the baseUrl, username and password.

```
String baseUrl =
    "https://localhost:8080/microblog/global/api/rest/2.4";
String username = "kenmei";
String password = "123456";

CommunoteClient client =
    new CommunoteClient(baseUrl, username, password);
```

### 2.2.1 Create Topic

This is an example to create a topic.

### Dependencies

```
com.communote.plugins.api.rest.client.resource.tag.TagResource;
com.communote.plugins.api.rest.client.resource.topic.CreateTopicParameter;
```

### Source Code

The title is mandatory to create a topic resource.

```
CreateTopicParameter createTopicParameter =
    new CreateTopicParameter();
createTopicParameter.setTitle("example");
TagResource tagResource = new TagResource();
tagResource.setDefaultName("tag");
createTopicParameter.setTags(new TagResource[] { tagResource });

Long topicId = client.createTopic(createTopicParameter);
```

### 2.2.2 Get Topic

This is an example to get a topic.

#### Dependency

```
com.communote.plugins.api.rest.client.resource.topic.GetTopicParameter;
```

#### Source Code

The topic identifier is a string because it can also be a alias. You must set the parameter `topicIdentifier=ALIAS` if the topic id is the alias of topic.

```
GetTopicParameter getTopicParameter = new GetTopicParameter();
// Is String because topic identifier can also be a alias
// only in combination with the parameter topicIdentifier=ALIAS
getTopicParameter.setTopicId(String.valueOf(topicId));

TopicResource topicResource = client.getTopic(getTopicParameter);
```

### 2.2.3 Edit Topic

This is an example to edit a topic.

#### Dependencies

```
com.communote.plugins.api.rest.client.resource.tag.TagResource;
com.communote.plugins.api.rest.client.resource.topic.EditTopicParameter;
```

#### Source Code

The topic identifier is a string because it can also be a alias. You must set the parameter `topicIdentifier=ALIAS` if the topic id is the alias of topic.

```
EditTopicParameter editTopicParameter = new EditTopicParameter();
editTopicParameter.setTopicId(String.valueOf(topicId));
editTopicParameter.setTitle("new example");
TagResource newTagResource = new TagResource();
newTagResource.setDefaultName("new tag");

topicId = client.editTopic(editTopicParameter);
```

### 2.2.4 List Topics

This is an example to get a list of topics.

#### Dependencies

```
com.communote.plugins.api.rest.client.resource.topic.ETopicListType;  
com.communote.plugins.api.rest.client.resource.topic.GetCollectionTopicParameter;  
com.communote.plugins.api.rest.client.resource.topic.TopicResource;
```

### Source Code

The `topicListType` is mandatory.

```
GetCollectionTopicParameter getCollectionTopicParameter =  
    new GetCollectionTopicParameter();  
getCollectionTopicParameter.setTopicListType(  
    ETopicListType.MANAGER);  
  
List<TopicResource> topicResources =  
    client.getCollectionTopic(getCollectionTopicParameter);
```

### 2.2.5 Delete Topic

This is an example to delete a topic.

#### Dependency

```
com.communote.plugins.api.rest.client.resource.topic.DeleteTopicParameter;
```

### Source Code

The topic identifier is a string because it can also be a alias. You must set the parameter `topicIdentifier=ALIAS` if the topic id is the alias of topic.

```
DeleteTopicParameter deleteTopicParameter =  
    new DeleteTopicParameter();  
deleteTopicParameter.setTopicId(String.valueOf(topicId));  
  
client.deleteTopic(deleteTopicParameter);
```

### 3 Usage of Resources

This is the Communote REST API for the communote microblogging system. The API provides a resources for the components of communote (e.g. Topics, Notes). An overview of all resources you can found under [Available Resource](#).

We recommend to use the Content-Type `application/json; charset=UTF-8` for request. Please read the section [Tunneling](#) before you start an request. (*Thats necessary because there is a distinctive different for using requests with HTTP-Method PUT and DELETE.*)

If the request was succeed, you get the following response format:

```
"status": "OK"
// returned resource as json or resources as json-array
"result": ...
// only if you request with a resource with HTTP-Methode LIST
"metaData":
{
    // number of all found elements
    "numberOfElements": "45"
    // there are more elements after the
    // choosen offset and maxCount
    "moreElementsAvailable": "true"
}
```

If the request fails, you get the following error response format:

```
"status": "ERROR"

// general error message
"message": "general message"

// only if you request with a resource with HTTP-Methode LIST
"errors":
{
    // mentioned incorrect element
    "cause": "element"
    // specific error message for this incorrect element
    "message": "element is incorrect"
}
```

As Response you get Content-Type `application/json` or Content-Type `text/html`. Content-Type `text/html` gives you an html-container who contain a global javascript variable with the result. The table below gives you an explanation for using extensions.

URL-Extension	Response Content-Type
<code>.../Resource/ID</code>	<code>application/json; charset=UTF-8</code>
<code>.../Resource/ID.json</code>	<code>application/json; charset=UTF-8</code>
<code>.../Resource/ID.html</code>	<code>text/html; charset=UTF-8</code>

## 3.1 Tunneling

Some servers or proxies don't allow the use of other HTTP methods than GET and POST, also many browsers only support GET and POST. To circumvent this problem it is possible to use a technique called **tunneling**.

Instead of sending the request using the intended method, for instance PUT, the request will be send as POST and the intended method will be set as GET parameter `_method`.

Please refer to the servers documentation to find out, if you have to use tunneling for your requests. This documentation was generated with tunneling enabled.

### 3.1.1 Usage of HTTP-Methods

Method	Tunneling Method	Tunneling Url
POST	POST	.../Resource/
PUT	POST	.../Resource/ID? _method=PUT
GET	GET	.../Resource/ID
LIST	GET	.../Resource/
DELETE	POST	.../Resource/ID? _method=DELETE



## 4 Available Resources

---

- Attachment Resource
  - Preview Resource
- Group Resource
  - Member Resource
- LastModificationDate Resource
- Note Resource
  - Favorite Resource
  - Like Resource
  - Property Resource
- Tag Resource
- TagStoreTagList Resource
  - TagStoreTag Resource
- TagSuggestionList Resource
- TimelineNote Resource
  - Property Resource
  - TimelineNoteDiscussion Resource
  - TimelineNoteTag Resource
- TimelineTag Resource
- TimelineTopic Resource
- TimelineUser Resource
- Topic Resource
  - ChildTopic Resource
  - ExternalObject Resource
  - Follow Resource
  - Property Resource
  - Right Resource
  - Role Resource
  - RoleBulkExternal Resource
    - RoleBulkExternalTopicRight Resource
  - RoleExternal Resource
- User Resource
  - ExternalLogin Resource

- [ExternalProperty Resource](#)
- [Follow Resource](#)
- [Image Resource](#)
- [NavigationItem Resource](#)
- [Observation Resource](#)
- [Property Resource](#)
- [UserPreference Resource](#)
  - [Preference Resource](#)

## 5 Attachment Resource and Subresources

---

### 5.1 Overview

- [Attachment Resource](#)
- [Preview Resource](#)

### 5.2 Resources

#### 5.2.1 Attachment Resource

This resource represents an attachment of a note.

#### Available HTTP-Methods for Attachment Resource

- [POST](#)
- [GET](#)
- [DELETE](#)

---

#### POST - Attachment Resource

By sending a HTTP-POST request with all mandatory attributes, the server **creates** a new attachment resource and returns a list of current uploaded attachments.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking post requests on your server.*

#### POST - Url

.../attachments/

#### POST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	<b>attachmentUploadSessionId</b> (mandatory)	Session identifier for session attribute.
boolean	isBase64	File attribute is base64 encoded.

#### POST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	fileName	This is the name of the file.
string	fileType	Is the type of the file might be represented by a file extension.

---

#### GET - Attachment Resource

By sending a HTTP-GET request, the server **returns** the requested attachment resource with the following attributes. To get the attachment as binary, set parameter `getAttachmentAsBinary=true`.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking get requests on your server.*

### GET - Url

`.../attachments/ID`

### GET - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
boolean	<code>getAttachmentAsBinary</code>	Returns the attachment as binary file. <b>Default Value</b> is false.

### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
long	<b>attachmentId</b> ( <i>mandatory</i> )	Is the Id of the attachment.
string	<code>contentId</code>	The content identifier. Depends on the associated repository identified by <code>repositoryConnectorId</code> . Since Communote version 3.3
long	<code>contentLength</code>	This is the size of the attachment measured in byte.
string	<code>fileName</code>	This is the name of the file.
string	<code>fileType</code>	Is the type of the file might be represented by a file extension.
long	<code>noteId</code>	Is the Id of the associated note. Since Communote version 3.3
string	<code>repositoryConnectorId</code>	The identifier of the repository the content is stored in. Since Communote version 3.3

---

### DELETE - Attachment Resource

#### DELETE - Url

`.../attachments/ID?_method=DELETE`

#### DELETE - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
------	------	-------------

long	attachmentId	Is the Id of the attachment.
------	--------------	------------------------------

#### 5.2.1.1 Preview Resource

This resource presents the preview for the given attachment

#### Available HTTP-Methods for Preview Resource

- [GET](#)

---

#### GET - Preview Resource

By sending a HTTP-GET request, the server **returns** a binary preview for the given attachment. Note, that currently only images supports previews.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking get requests on your server.*

#### GET - Url

.../attachments/ID/previews/ID

#### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
long	<b>previewId</b> ( <i>mandatory</i> )	Is the Id of the attachment.

## 6 Note Resource and Subresources

---

### 6.1 Overview

- [Note Resource](#)
  - [Favorite Resource](#)
  - [Like Resource](#)
  - [Property Resource](#)

### 6.2 Resources

#### 6.2.1 Note Resource

This is the resource for a note.

##### Available HTTP-Methods for Note Resource

- [POST](#)
- [PUT](#)
- [GET](#)
- [DELETE](#)

---

#### POST - Note Resource

By sending a HTTP-POST request with all mandatory attributes, the server **creates** a new note resource and returns the noteId.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking post requests on your server.*

#### POST - Url

.../notes/

#### POST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
boolean	isHtml	Depracated. Use <a href="#">TimelineNote Resource</a> List instead.
integer	maxTextLength	Depracated. Use <a href="#">TimelineNote Resource</a> List instead.
boolean	publish	Should note publish or is an auto saved note draft. <b>Default Value</b> is true.
long	autosaveNoteId	Identifier of auto save note draft.
string[]	crossPostTopicAliases	Topic aliases to save note in different topics as cross post.

string	attachmentUploadSessionId	Session identifier of an session attribute witch contains attachmentIds to upload.
--------	---------------------------	--

## POST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
long[]	attachmentIds	Identifier of the note's attachments
boolean	isDirectMessage	Indicates whether the note is a direct message or a normal message. <b>Default Value</b> is false.
long	noteId	Identifier of the note
long	noteVersion	Current version of note.
long	parentNoteId	The identifier of the parent note.
Property[]	properties	Properties of note. Properties are key value pairs.
Tag[]	tags	Tags with witch the note is tagged.
string	text	The message text of the note.
long	<b>topicId</b> ( <i>mandatory</i> )	Is the identifier of the topic where this note will be published
string[]	usersToNotify	Specifies the aliases of users which are @-addressed in this message.

---

## PUT - Note Resource

By sending a HTTP-PUT request with edited attributes, the server **edits** the note resource and returns the noteId.

### PUT - Url

.../notes/ID?\_method=PUT

### PUT - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
boolean	isHtml	Depracated. Use <a href="#">TimelineNote Resource</a> List instead.
integer	maxTextLength	Depracated. Use <a href="#">TimelineNote Resource</a> List instead.
boolean	publish	Should note publish or is an auto saved note draft. <b>Default Value</b> is true.
long	autosaveNoteId	Identifier of auto save note draft.

string[]	crossPostTopicAliases	Topic aliases to save note in different topics as cross post.
string	attachmentUploadSessionId	Session identifier of an session attribute witch contains attachmentIds to upload.

### PUT - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
long[]	attachmentIds	Identifier of the note's attachments
long	<b>noteId</b> ( <i>mandatory</i> )	Identifier of the note
long	noteVersion	Current version of note.
Property[]	properties	Properties of note. Properties are key value pairs.
Tag[]	tags	Tags with witch the note is tagged.
string	text	The message text of the note.
long	<b>topicId</b> ( <i>mandatory</i> )	Is the identifier of the topic where this note will be published
string[]	usersToNotify	Specifies the aliases of users which are @-addressed in this message.

---

### GET - Note Resource

By sending a HTTP-GET request, the server **returns** the requested note resource with the following attributes.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking get requests on your server.*

#### GET - Url

.../notes/ID

#### GET - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
boolean	filterHtml	Depracated. Use <a href="#">TimelineNote Resource</a> List instead.

### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
long[]	attachmentIds	Identifier of the note's attachments



Attachment[]	attachments	Contains all attachment resources of the note.
date	creationDate	Creation date of a note
integer	discussionDepth	It the amount of replies within a discussion.
long	discussionId	Is the id of the whole discussion.
string	discussionPath	Returns the path of the discussions.
string	firstName	This is the firstname of the author.
string[]	followedItems	Indicates whether the note is part of a followed author, topic or discussion.
boolean	isDirectMessage	Indicates whether the note is a direct message or a normal message. <b>Default Value</b> is false.
boolean	isFavorite	Specifies whether the note is marked as a favorite.
boolean	isLike	Specifies whether the note is marked as a like.
boolean	isMentionDiscussionAuthors	Indicates that this note addresses all participants of the discussion this note is part of.
boolean	isMentionTopicAuthors	Indicates that this note addresses all users with at least one note within the topic.
boolean	isMentionTopicManagers	Indicates that this note addresses all managers of the the topic.
boolean	isMentionTopicReaders	Indicates that this note addresses all readers of the topic.
boolean	isNoteForMe	Indicates whether the note is a addressed to the current user.
date	lastModificationDate	Date of the last modification
string	lastName	This is the surname of the author.
long	<b>notedId</b> ( <i>mandatory</i> )	Identifier of the note
long	noteVersion	Current version of note.
integer	numberOfComments	Number of comments (also called replys or children).
long	parentNotedId	The identifier of the parent note.
Tag[]	tags	Tags with witch the note is tagged.
string	text	The message text of the note.
long	topicId	Is the identifier of the topic where this note will be published
string	topicName	This is the name of the topic in witch the note was posted.
long	userId	It the identifier of the author.

string	userName	This is the alias of the author.
string[]	usersToNotify	Specifies the aliases of users which are @-addressed in this message.
long[]	usersToNotifyIds	Specifies the user ids of users which are @-addressed in this message. Since Communote version 3.3.

---

## DELETE - Note Resource

### DELETE - Url

.../notes/ID?\_method=DELETE

### DELETE - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
long	<b>noteId</b> ( <i>mandatory</i> )	Identifier of the note

#### 6.2.1.1 Favorite Resource

Resource for setting a favorite. It is a sub resource of a note.

### Available HTTP-Methods for Favorite Resource

- [POST](#)
- [LIST](#)
- [DELETE](#)

---

## POST - Favorite Resource

By sending a HTTP-POST request with all mandatory attributes, the server sets the favorite status for specified note and returns null.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking post requests on your server.*

### POST - Url

.../notes/ID/favorites/

### POST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
boolean	favorite	favorite status of the note for the user. <b>Default Value</b> is false.

---

## LIST - Favorite Resource

**LIST - Url**

.../notes/ID/favorites/

**LIST - Attributes**

Attributes are part of the entity send or requested.

Type	Name	Description
boolean	favorite	favorite status of the note for the user. <b>Default Value</b> is false.

**DELETE - Favorite Resource****DELETE - Url**

.../notes/ID/favorites/ID?\_method=DELETE

**DELETE - Attributes**

Attributes are part of the entity send or requested.

Type	Name	Description
------	------	-------------

**6.2.1.2 Like Resource**

Resource for setting a like. It is a sub resource of a note.

**Available HTTP-Methods for Like Resource**

- [POST](#)
- [LIST](#)
- [DELETE](#)

**POST - Like Resource**

By sending a HTTP-POST request with all mandatory attributes, the server sets the like status for specified note and returns null.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking post requests on your server.*

**POST - Url**

.../notes/ID/likes/

**POST - Attributes**

Attributes are part of the entity send or requested.

Type	Name	Description
boolean	like	like status of the note for the user. <b>Default Value</b> is false.

**LIST - Like Resource****LIST - Url**

.../notes/ID/likes/

**LIST - Attributes**

Attributes are part of the entity send or requested.

Type	Name	Description
boolean	like	like status of the note for the user. <b>Default Value</b> is false.
TimelineUser[]	users	Specifies the users who likes the note inclusive the current user when he likes the note.

**DELETE - Like Resource****DELETE - Url**

.../notes/ID/likes/ID?\_method=DELETE

**DELETE - Attributes**

Attributes are part of the entity send or requested.

Type	Name	Description
------	------	-------------

**6.2.1.3 Property Resource**

Resource for setting a property. It is a sub resource of a note.

**Available HTTP-Methods for Property Resource**

- [POST](#)
- [GET](#)
- [LIST](#)

**POST - Property Resource**

By sending a HTTP-POST request with all mandatory attributes, the server **creates** a new property resource and returns null.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking post requests on your server.*

**POST - Url**

.../notes/ID/propertys/

**POST - Attributes**

Attributes are part of the entity send or requested.

Type	Name	Description
------	------	-------------

string	<b>key</b> ( <i>mandatory</i> )	Key of the property.
string	<b>keyGroup</b> ( <i>mandatory</i> )	KeyGroup of the property.
string	value	Value of property for specific key.

---

### GET - Property Resource

#### GET - Url

.../notes/ID/propertys/ID

#### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	<b>key</b> ( <i>mandatory</i> )	Key of the property.
string	<b>keyGroup</b> ( <i>mandatory</i> )	KeyGroup of the property.
long	propertyId	KeyGroup of the property.
string	value	Value of property for specific key.

---

### LIST - Property Resource

#### LIST - Url

.../notes/ID/propertys/

#### LIST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	key	Key of the property.
string	keyGroup	KeyGroup of the property.

## 7 Tag Resource and Subresources

---

### 7.1 Overview

- [Tag Resource](#)

### 7.2 Resources

#### 7.2.1 Tag Resource

This resource represents tag inside of an tag store.

##### Available HTTP-Methods for Tag Resource

- [PUT](#)
- [GET](#)
- [LIST](#)
- [DELETE](#)

---

#### PUT - Tag Resource

##### PUT - Url

.../tags/ID?\_method=PUT

##### PUT - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	<b>defaultName</b> ( <i>mandatory</i> )	This is the name of tag.
string	description	This is the description of tag.
string	languageCode	This is the language code of tag.
string	name	This is the name of tag.
boolean	renameTag	If set, it is only possible to edit the default name. Also only managers and system users may use this.
long	tagId	Is the identifier of the tag.
string	tagStoreAlias	This is the identifier of the tagstore.
string	tagStoreTagId	This is the identifier of tag inside tagstore.
string	tagStoreType	Type of tag store (BLOG, NOTE, ENTITY). Default type is NOTE.

---

#### GET - Tag Resource

By sending a HTTP-GET request, the server **returns** the requested the specified tag resource. There are two ways to get the tag. When the tag identifier in the url was set, the api search for this tag in the database. But when the tag identifier is set 0 then tagStoreTagId and tagStoreAlias must been specified to find the tag.

Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking get requests on your server.

### GET - Url

.../tags/ID

### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	defaultName	This is the name of tag.
string	description	This is the description of tag.
boolean	isFollow	Indicates whether the topic is followed by the user.
string	languageCode	This is the language code of tag.
string	name	This is the name of tag.
long	tagId	Is the identifier of the tag.
string	tagStoreAlias	This is the identifier of the tagstore.
string	tagStoreTagId	This is the identifier of tag inside tagstore.

---

### LIST - Tag Resource

### LIST - Url

.../tags/

### LIST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
boolean	isFollow	Indicates whether the topic is followed by the user.

---

### DELETE - Tag Resource

### DELETE - Url

.../tags/ID?\_method=DELETE

### DELETE - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
long	newTagId	Used, when a tag is removed to assing all notes to the new tag.

long	tagId	Is the identifier of the tag.
------	-------	-------------------------------



## 8 TagStoreTagList Resource and Subresources

---

### 8.1 Overview

- [TagStoreTagList Resource](#)
- [TagStoreTag Resource](#)

### 8.2 Resources

#### 8.2.1 TagStoreTagList Resource

This resource represents an list of tagstores with taglists.

#### Available HTTP-Methods for TagStoreTagList Resource

- [GET](#)
- [LIST](#)

---

#### GET - TagStoreTagList Resource

##### GET - Url

.../tagStoreTagLists/ID

##### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	tagStoreAlias	This is the alias of the tagStore. If tag store alias was not found then the internal note tagStore is the fallback.
TagStoreTag[]	tags	This is the name of the tagStore.

---

#### LIST - TagStoreTagList Resource

##### LIST - Url

.../tagStoreTagLists/

##### LIST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	tagStoreAlias	This is the alias of the tagStore. If tag store alias was not found then the internal note tagStore is the fallback.
TagStoreTag[]	tags	This is the name of the tagStore.

### 8.2.1.1 TagStoreTag Resource

This resource represents an tag of an tag store.

#### Available HTTP-Methods for TagStoreTag Resource

- [GET](#)
- [LIST](#)

---

#### GET - TagStoreTag Resource

##### GET - Url

.../tagStoreTagLists/ID/tagStoreTags/ID

##### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	description	This is the description of an tag.
string	name	This is the name of an tag.
long	tagId	Is the Id of an tag.
string	tagStoreTagId	Is the Id of an tag in parent tagStore.

---

#### LIST - TagStoreTag Resource

##### LIST - Url

.../tagStoreTagLists/ID/tagStoreTags/

##### LIST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	description	This is the description of an tag.
string	name	This is the name of an tag.
long	tagId	Is the Id of an tag.
string	tagStoreTagId	Is the Id of an tag in parent tagStore.

## 9 TagSuggestionList Resource and Subresources

### 9.1 Overview

- [TagSuggestionList Resource](#)

### 9.2 Resources

#### 9.2.1 TagSuggestionList Resource

This resource represents an list of tagstores with taglists.

#### Available HTTP-Methods for TagSuggestionList Resource

- [POST](#)
- [LIST](#)

---

#### POST - TagSuggestionList Resource

##### POST - Url

.../tagSuggestionLists/

##### POST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
integer	offset	Specifies the offset of the list to return results for. <b>Default Value</b> is 0
integer	maxCount	Specifies the maximum amount of items in the resulting list. <b>Default Value</b> is 10. <b>Note:</b> Using it for retrieving the topic list the following parameter is required: <code>topicListType=LAST_MODIFIED!</code>
integer	checkAtLeastMoreResults	With this parameter it is possible to define a number of additional entities, which should be loaded when requesting data. These will be used to build information for paging, endless scrolling etc..
string[]	f_contentIds	Filters for notes with attachments with one of the given content identifiers. Since Communote version 3.3.
long	f_discussionId	Filters notes which are part of the specified discussion. If the discussion identifier is set the sort mode change from chronological (newest first) to threaded view style. Additional the discussion depth is set.

boolean	f_showFollowedItems	Filters notes which are followed by the user.
boolean	f_showNotesForMe	Filters notes which are addressed for the current user.
long[]	f_topicIds	Filters for notes with a topic being part of the given identifiers.
string[]	f_topicAliases	Filters for notes with topics being part of the given aliases.
long[]	f_topicIdsToExclude	Filters for notes NOT being part of the given topic identifiers.
long[]	f_tagIds	Filters for notes having all the tags assigned. The tag identifier refers to the tag database identifier.
string	f_tagPrefix	Filters for notes containing at least one tag starting with the given string.
long[]	f_userIds	Filters for notes with an author matching one of the given user identifiers.
string[]	f_userAliases	Filters for notes with an author matching one of the given user aliases.
date	f_startDate	Filters for notes being written after the specified date.
date	f_endDate	Filters for notes being written before or on the given date. <b>Note:</b> The specified day is included. If this day should be excluded, subtract 24 hours (86.400.000 ms) from f_endDate.
string	f_userSearchString	Filters for notes with an authors alias, last name or first name matching one of the given search string.
string	f_noteSearchString	Filters for notes containing the given string. <b>Note:</b> The results may vary depending on the Communote database used.
string	f_fullTextSearchString	Filters for notes where either the note content, the tags of the note or the name of the author contains the given string.
long	f_parentNoteId	Filters for notes with the specified parent's note id and the parent note itself.
string[]	f_repositoryIds	Filters for notes with attachments with the given repository identifiers. The order must match to the given f_contentIds parameter. Since Communote version 3.3.

boolean	f_showDirectMessages	Filters for notes which are direct messages.
boolean	f_showFavorites	Filters for notes which are marked as favorite.
boolean	f_includeComments	Specifies whether comments (or answers) of notes are included in the resulting list.
string	f_propertyFilter	Array of properties to filter for. The array is build like this: <i>FILTER_1, FILTER_2, ..., FILTER_3</i> where every FILTER looks like <code>&amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Example&amp;#34;,&amp;#34;value&amp;#34;:&amp;#34;[0:Property, 1:Group, 2:Key, 3:Value, 4:MatchMode, 5:Negate (Optional)]&amp;#125;,&amp;#34;whereas Property can be one of Note, Entity, Dummy and MatchMode can be one of EQUALS, STARTS_WITH, ENDS_WITH, CONTAINS, NOT_EQUALS. Negate is optional (default = false) can be true or false. <b>Note:</b> Every filter value has to be escaped with double quotes &amp;#34;. <b>Example:</b> <code>&amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Image&amp;#34;,&amp;#34;value&amp;#34;:&amp;#34;[&amp;#34;Note&amp;#34;,&amp;#34;com.communit&amp;#34;,&amp;#34;co</code></code>
string	f_topicAccess	Filter for notes which can be accessed at least by the given access level. <b>Constant Values</b> manager - Lists all topics of which the user is the manager. read - Lists all topics of which the user has read access. write - Lists all topics of which the user has write access.
boolean	f_showOnlyDiscussions	Show only the discussion.
boolean	f_showOnlyRootNotes	Show only root notes.
boolean	f_showDiscussionParticipation	Filter for discussions, the current user has participated in.
long	f_targetTopicId	Id of the topic to filter for.
long	f_noteId	Id of the note to filter for.
boolean	f_topicRelated	Specifies whether the results have to be topic/utp related.
long	f_retrieveOnlyNotesBeforeId	Only retrieve notes with a smaller id. Can only be used in conjunction with f_retrieveOnlyNotesBeforeDate.
long	f_retrieveOnlyNotesBeforeDate	Only retrieve notes created before this date. Can only be used in conjunction with f_retrieveOnlyNotesBeforeId.

long	f_retrieveOnlyNotesAfterId	Only retrieve notes with a later id. Can only be used in conjunction with f_retrieveOnlyNotesAfterDate.
long	f_retrieveOnlyNotesAfterDate	Only retrieve notes created after this date. Can only be used in conjunction with f_retrieveOnlyNotesAfterId.

## LIST - TagSuggestionList Resource

### LIST - Url

.../tagSuggestionLists/

### LIST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	tagStoreType	Type of tag store (BLOG, NOTE, ENTITY). Default type is NOTE.
string[]	f_suggestionAliases	Filter suggestion aliases.
string[]	f_suggestionProviderAliases	Filter suggestion provider aliases.
boolean	assignedTagsOnly	Filter or assign tags.
integer	offset	Specifies the offset of the list to return results for. <b>Default Value</b> is 0
integer	maxCount	Specifies the maximum amount of items in the resulting list. <b>Default Value</b> is 10. <b>Note:</b> Using it for retrieving the topic list the following parameter is required: <code>topicListType=LAST_MODIFIED!</code>
integer	checkAtLeastMoreResults	With this parameter it is possible to define a number of additional entities, which should be loaded when requesting data. These will be used to build information for paging, endless scrolling etc..
string[]	f_contentIds	Filters for notes with attachments with one of the given content identifiers. Since Communote version 3.3.
long	f_discussionId	Filters notes which are part of the specified discussion. If the discussion identifier is set the sort mode change from chronological (newest first) to threaded view style. Additionally the discussion depth is set.

boolean	f_showFollowedItems	Filters notes which are followed by the user.
boolean	f_showNotesForMe	Filters notes which are addressed for the current user.
long[]	f_topicIds	Filters for notes with a topic being part of the given identifiers.
string[]	f_topicAliases	Filters for notes with topics being part of the given aliases.
long[]	f_topicIdsToExclude	Filters for notes NOT being part of the given topic identifiers.
long[]	f_tagIds	Filters for notes having all the tags assigned. The tag identifier refers to the tag database identifier.
string	f_tagPrefix	Filters for notes containing at least one tag starting with the given string.
long[]	f_userIds	Filters for notes with an author matching one of the given user identifiers.
string[]	f_userAliases	Filters for notes with an author matching one of the given user aliases.
date	f_startDate	Filters for notes being written after the specified date.
date	f_endDate	Filters for notes being written before or on the given date. <b>Note:</b> The specified day is included. If this day should be excluded, subtract 24 hours (86.400.000 ms) from f_endDate.
string	f_userSearchString	Filters for notes with an authors alias, last name or first name matching one of the given search string.
string	f_noteSearchString	Filters for notes containing the given string. <b>Note:</b> The results may vary depending on the Communote database used.
string	f_fullTextSearchString	Filters for notes where either the note content, the tags of the note or the name of the author contains the given string.
long	f_parentNoteId	Filters for notes with the specified parent's note id and the parent note itself.
string[]	f_repositoryIds	Filters for notes with attachments with the given repository identifiers. The order must match to the given f_contentIds parameter. Since Communote version 3.3.

boolean	f_showDirectMessages	Filters for notes which are direct messages.
boolean	f_showFavorites	Filters for notes which are marked as favorite.
boolean	f_includeComments	Specifies whether comments (or answers) of notes are included in the resulting list.
string	f_propertyFilter	Array of properties to filter for. The array is build like this: <i>FILTER_1, FILTER_2, ..., FILTER_3</i> where every FILTER looks like <code>&amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Example&amp;#34;,&amp;#34;value&amp;#34;:&amp;#34;[0:Property, 1:Group, 2:Key, 3:Value, 4:MatchMode, 5:Negate (Optional)]&amp;#125;,&amp;#34;whereas Property can be one of Note, Entity, Dummy and MatchMode can be one of EQUALS, STARTS_WITH, ENDS_WITH, CONTAINS, NOT_EQUALS. Negate is optional (default = false) can be true or false. <b>Note:</b> Every filter value has to be escaped with double quotes &amp;#34;. <b>Example:</b> <code>&amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Image&amp;#34;,&amp;#34;value&amp;#34;:&amp;#34;[&amp;#34;Note&amp;#34;,&amp;#34;com.communit&amp;#34;,&amp;#34;co</code></code>
string	f_topicAccess	Filter for notes which can be accessed at least by the given access level. <b>Constant Values</b> manager - Lists all topics of which the user is the manager. read - Lists all topics of which the user has read access. write - Lists all topics of which the user has write access.
boolean	f_showOnlyDiscussions	Show only the discussion.
boolean	f_showOnlyRootNotes	Show only root notes.
boolean	f_showDiscussionParticipation	Filter for discussions, the current user has participated in.
long	f_targetTopicId	Id of the topic to filter for.
long	f_noteId	Id of the note to filter for.
boolean	f_topicRelated	Specifies whether the results have to be topic/utp related.
long	f_retrieveOnlyNotesBeforeId	Only retrieve notes with a smaller id. Can only be used in conjunction with f_retrieveOnlyNotesBeforeDate.
long	f_retrieveOnlyNotesBeforeDate	Only retrieve notes created before this date. Can only be used in conjunction with f_retrieveOnlyNotesBeforeId.



long	f_retrieveOnlyNotesAfterId	Only retrieve notes with a later id. Can only be used in conjunction with f_retrieveOnlyNotesAfterDate.
long	f_retrieveOnlyNotesAfterDate	Only retrieve notes created after this date. Can only be used in conjunction with f_retrieveOnlyNotesAfterId.

### LIST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	alias	Alias of suggestion.
string	name	Name of suggestion.
string	providerAlias	Alias of suggestion provider.
Tag[]	tags	Tags of suggestion.

## 10 TimelineNote Resource and Subresources

---

### 10.1 Overview

- [TimelineNote Resource](#)
  - [Property Resource](#)
  - [TimelineNoteDiscussion Resource](#)
  - [TimelineNoteTag Resource](#)

### 10.2 Resources

#### 10.2.1 TimelineNote Resource

This resource represents the timeline of notes.

#### Available HTTP-Methods for TimelineNote Resource

- [POST](#)
- [PUT](#)
- [GET](#)
- [LIST](#)

---

#### POST - TimelineNote Resource

By sending a HTTP-POST request, the server **returns** a list of all requested timeline note resources with the following attributes. Additional there are two meta datas in the response, use it in case the query parameters exceed the supported length. First with the number of all found elements and second whether there are more available elements.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking post requests on your server.*

#### POST - Url

.../timelineNotes/

#### POST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
boolean	filterHtml	Specifies whether the timeline resources response is formatted as plain text or with HTML tags.
integer	maxTextLength	Specifies the length of the note text to retrieve a list of timeline resources with a cut off text for better list representation.
integer	offset	Specifies the offset of the list to return results for. <b>Default Value</b> is 0

integer	maxCount	Specifies the maximum amount of items in the resulting list. <b>Default Value</b> is 10. <b>Note:</b> Using it for retrieving the topic list the following parameter is required: <code>topicListType=LAST_MODIFIED!</code>
integer	checkAtLeastMoreResults	With this parameter it is possible to define a number of additional entities, which should be loaded when requesting data. These will be used to build information for paging, endless scrolling etc..
string[]	f_contentIds	Filters for notes with attachments with one of the given content identifiers. Since Communote version 3.3.
long	f_discussionId	Filters notes which are part of the specified discussion. If the discussion identifier is set the sort mode change from chronological (newest first) to threaded view style. Additional the discussion depth is set.
boolean	f_showFollowedItems	Filters notes which are followed by the user.
boolean	f_showNotesForMe	Filters notes which are addressed for the current user.
long[]	f_topicIds	Filters for notes with a topic being part of the given identifiers.
string[]	f_topicAliases	Filters for notes with topics being part of the given aliases.
long[]	f_topicIdsToExclude	Filters for notes NOT being part of the given topic identifiers.
long[]	f_tagIds	Filters for notes having all the tags assigned. The tag identifier refers to the tag database identifier.
string	f_tagPrefix	Filters for notes containing at least one tag starting with the given string.
long[]	f_userIds	Filters for notes with an author matching one of the given user identifiers.
string[]	f_userAliases	Filters for notes with an author matching one of the given user aliases.
date	f_startDate	Filters for notes being written after the specified date.

date	f_endDate	Filters for notes being written before or on the given date. <b>Note:</b> The specified day is included. If this day should be excluded, subtract 24 hours (86.400.000 ms) from f_endDate.
string	f_userSearchString	Filters for notes with an authors alias, last name or first name matching one of the given search string.
string	f_noteSearchString	Filters for notes containing the given string. <b>Note:</b> The results may vary depending on the Communote database used.
string	f_fullTextSearchString	Filters for notes where either the note content, the tags of the note or the name of the author contains the given string.
long	f_parentNoteId	Filters for notes with the specified parent's note id and the parent note itself.
string[]	f_repositoryIds	Filters for notes with attachments with the given repository identifiers. The order must match to the given f_contentIds parameter. Since Communote version 3.3.
boolean	f_showDirectMessages	Filters for notes which are direct messages.
boolean	f_showFavorites	Filters for notes which are marked as favorite.
boolean	f_includeComments	Specifies whether comments (or answers) of notes are included in the resulting list.
string	f_propertyFilter	Array of properties to filter for. The array is build like this: <i>FILTER_1, FILTER_2, ..., FILTER_3</i> where every FILTER looks like <code>&amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Example&amp;#34;,&amp;#34;value&amp;#34;:[0:Property, 1:Group, 2:Key, 3:Value, 4:MatchMode, 5:Negate (Optional)]&amp;#125;,</code> whereas <i>Property</i> can be one of <i>Note</i> , <i>Entity</i> , <i>Dummy</i> and <i>MatchMode</i> can be one of <i>EQUALS</i> , <i>STARTS_WITH</i> , <i>ENDS_WITH</i> , <i>CONTAINS</i> , <i>NOT_EQUALS</i> . <i>Negate</i> is optional (default = false) can be <i>true</i> or <i>false</i> . <b>Note:</b> Every filter value has to be escaped with double quotes &#34;. <b>Example:</b> <code>&amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Image&amp;#34;,&amp;#34;value&amp;#34;:[&amp;#34;Note&amp;#34;,&amp;#34;com.communote&amp;#34;,&amp;#34;co</code>

string	f_topicAccess	Filter for notes which can be accessed at least by the given access level. <b>Constant Values</b> manager - Lists all topics of which the user is the manager. read - Lists all topics of which the user has read access. write - Lists all topics of which the user has write access.
boolean	f_showOnlyDiscussions	Show only the discussion.
boolean	f_showOnlyRootNotes	Show only root notes.
boolean	f_showDiscussionParticipation	Filter for discussions, the current user has participated in.
long	f_targetTopicId	Id of the topic to filter for.
long	f_noteId	Id of the note to filter for.
boolean	f_topicRelated	Specifies whether the results have to be topic/utp related.
long	f_retrieveOnlyNotesBeforeId	Only retrieve notes with a smaller id. Can only be used in conjunction with f_retrieveOnlyNotesBeforeDate.
long	f_retrieveOnlyNotesBeforeDate	Only retrieve notes created before this date. Can only be used in conjunction with f_retrieveOnlyNotesBeforeId.
long	f_retrieveOnlyNotesAfterId	Only retrieve notes with a later id. Can only be used in conjunction with f_retrieveOnlyNotesAfterDate.
long	f_retrieveOnlyNotesAfterDate	Only retrieve notes created after this date. Can only be used in conjunction with f_retrieveOnlyNotesAfterId.

## POST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
Attachment[]	attachments	Contains all attachment resources.
TimelineUser	author	Contains the author of note.
date	creationDate	The date of creation of note.
TimelineNoteDiscussion	discussion	Contains information of the current discussion, where the note is parent or discussion part.
boolean	isDirectMessage	Indicates whether the note is a direct message or a normal message.

long	noteld	Is the identifier of an note.
integer	numberOfChildNotes	Number of all child notes.
integer	numberOfLikes	Number of users who like this note.
long	parentNoteld	Note identifier of the parent note. Is null when note is parent.
Property[]	properties	Properties of note. Properties are key value pairs.
string[]	rights	Specifies the possibilities what the user can do with this note. <b>Constant Values</b> CAN_EDIT - <i>Mark that the user can edit this note.</i> CAN_REPLY - <i>Mark that the user can reply this note.</i> CAN_DELETE - <i>Mark that the user can delete this note.</i>
string	shortText	The short message of note.
TimelineNoteTag[]	tags	Tags of the note.
string	text	The message of the note.
TimelineTopic	topic	Contains the topic where the not was posted.
string[]	userNoteProperties	Specifies properties of the current user. Multiple markers can be set. <b>Constant Values</b> FAV - <i>Mark for note is favor of current user.</i> LIKE - <i>Mark that the current user like this note.</i> NOTIFY - <i>Mark that the current user is notify in this note.</i>
TimelineUser[]	usersToNotify	Contains all notified users.

---

### PUT - TimelineNote Resource

#### PUT - Url

.../timelineNotes/ID?\_method=PUT

#### PUT - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
Property[]	properties	Properties of note. Properties are key value pairs.

---

### GET - TimelineNote Resource

By sending a HTTP-GET request, the server **returns** the requested timeline note resource with the following attributes.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking get requests on your server.*

### GET - Url

.../timelineNotes/ID

### GET - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
boolean	filterHtml	Specifies whether the timeline resources response is formatted as plain text or with HTML tags.
integer	maxTextLength	Specifies the length of the note text to retrieve a list of timeline resources with a cut off text for better list representation.

### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
Attachment[]	attachments	Contains all attachment resources.
TimelineUser	author	Contains the author of note.
date	creationDate	The date of creation of note.
TimelineNoteDiscussion	discussion	Contains information of the current discussion, where the note is parent or discussion part.
boolean	isDirectMessage	Indicates whether the note is a direct message or a normal message.
boolean	isMentionDiscussionAuthors	Indicates that this note addresses all participants of the discussion this note is part of.
boolean	isMentionTopicAuthors	Indicates that this note addresses all users with at least one note within the topic.
boolean	isMentionTopicManagers	Indicates that this note addresses all managers of the the topic.
boolean	isMentionTopicReaders	Indicates that this note addresses all readers of the topic.
long	<b>noteId</b> ( <i>mandatory</i> )	Is the identifier of an note.
integer	numberOfChildNotes	Number of all child notes.
integer	numberOfLikes	Number of users who like this note.

long	parentNoteId	Note identifier of the parent note. Is null when note is parent.
string[]	rights	Specifies the possibilities what the user can do with this note. <b>Constant Values</b> CAN_EDIT - Mark that the user can edit this note. CAN_REPLY - Mark that the user can reply this note. CAN_DELETE - Mark that the user can delete this note.
string	shortText	The short message of note.
TimelineNoteTag[]	tags	Tags of the note.
string	text	The message of the note.
TimelineTopic	topic	Contains the topic where the note was posted.
string[]	userNoteProperties	Specifies properties of the current user. Multiple markers can be set. <b>Constant Values</b> FAV - Mark for note is favor of current user. LIKE - Mark that the current user like this note. NOTIFY - Mark that the current user is notify in this note.
TimelineUser[]	usersToNotify	Contains all notified users.

## LIST - TimelineNote Resource

By sending a HTTP-GET request, the server **returns** a list of all requested timeline note resources with the following attributes. Additional there are two meta datas in the response. First with the number of all found elements and second whether there are more available elements.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking list requests on your server.*

### LIST - Url

.../timelineNotes/

### LIST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
boolean	filterHtml	Specifies whether the timeline resources response is formatted as plain text or with HTML tags.
integer	maxTextLength	Specifies the length of the note text to retrieve a list of timeline resources with a cut off text for better list representation.



integer	offset	Specifies the offset of the list to return results for. <b>Default Value</b> is 0
integer	maxCount	Specifies the maximum amount of items in the resulting list. <b>Default Value</b> is 10. <b>Note:</b> Using it for retrieving the topic list the following parameter is required: <code>topicListType=LAST_MODIFIED!</code>
integer	checkAtLeastMoreResults	With this parameter it is possible to define a number of additional entities, which should be loaded when requesting data. These will be used to build information for paging, endless scrolling etc..
string[]	f_contentIds	Filters for notes with attachments with one of the given content identifiers. Since Communote version 3.3.
long	f_discussionId	Filters notes which are part of the specified discussion. If the discussion identifier is set the sort mode change from chronological (newest first) to threaded view style. Additional the discussion depth is set.
boolean	f_showFollowedItems	Filters notes which are followed by the user.
boolean	f_showNotesForMe	Filters notes which are addressed for the current user.
long[]	f_topicIds	Filters for notes with a topic being part of the given identifiers.
string[]	f_topicAliases	Filters for notes with topics being part of the given aliases.
long[]	f_topicIdsToExclude	Filters for notes NOT being part of the given topic identifiers.
long[]	f_tagIds	Filters for notes having all the tags assigned. The tag identifier refers to the tag database identifier.
string	f_tagPrefix	Filters for notes containing at least one tag starting with the given string.
long[]	f_userIds	Filters for notes with an author matching one of the given user identifiers.
string[]	f_userAliases	Filters for notes with an author matching one of the given user aliases.
date	f_startDate	Filters for notes being written after the specified date.

date	f_endDate	Filters for notes being written before or on the given date. <b>Note:</b> The specified day is included. If this day should be excluded, subtract 24 hours (86.400.000 ms) from f_endDate.
string	f_userSearchString	Filters for notes with an authors alias, last name or first name matching one of the given search string.
string	f_noteSearchString	Filters for notes containing the given string. <b>Note:</b> The results may vary depending on the Communote database used.
string	f_fullTextSearchString	Filters for notes where either the note content, the tags of the note or the name of the author contains the given string.
long	f_parentNoteId	Filters for notes with the specified parent's note id and the parent note itself.
string[]	f_repositoryIds	Filters for notes with attachments with the given repository identifiers. The order must match to the given f_contentIds parameter. Since Communote version 3.3.
boolean	f_showDirectMessages	Filters for notes which are direct messages.
boolean	f_showFavorites	Filters for notes which are marked as favorite.
boolean	f_includeComments	Specifies whether comments (or answers) of notes are included in the resulting list.
string	f_propertyFilter	Array of properties to filter for. The array is build like this: <i>FILTER_1, FILTER_2, ..., FILTER_3</i> where every FILTER looks like <code>&amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Example&amp;#34;,&amp;#34;value&amp;#34;:[0:Property, 1:Group, 2:Key, 3:Value, 4:MatchMode, 5:Negate (Optional)]&amp;#125;,</code> whereas <i>Property</i> can be one of <i>Note</i> , <i>Entity</i> , <i>Dummy</i> and <i>MatchMode</i> can be one of <i>EQUALS</i> , <i>STARTS_WITH</i> , <i>ENDS_WITH</i> , <i>CONTAINS</i> , <i>NOT_EQUALS</i> . <i>Negate</i> is optional (default = false) can be <i>true</i> or <i>false</i> . <b>Note:</b> Every filter value has to be escaped with double quotes &#34;. <b>Example:</b> <code>&amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Image&amp;#34;,&amp;#34;value&amp;#34;:[&amp;#34;Note&amp;#34;,&amp;#34;com.communote&amp;#34;,&amp;#34;co</code>

string	f_topicAccess	Filter for notes which can be accessed at least by the given access level. <b>Constant Values</b> manager - Lists all topics of which the user is the manager. read - Lists all topics of which the user has read access. write - Lists all topics of which the user has write access.
boolean	f_showOnlyDiscussions	Show only the discussion.
boolean	f_showOnlyRootNotes	Show only root notes.
boolean	f_showDiscussionParticipation	Filter for discussions, the current user has participated in.
long	f_targetTopicId	Id of the topic to filter for.
long	f_noteId	Id of the note to filter for.
boolean	f_topicRelated	Specifies whether the results have to be topic/utp related.
long	f_retrieveOnlyNotesBeforeId	Only retrieve notes with a smaller id. Can only be used in conjunction with f_retrieveOnlyNotesBeforeDate.
long	f_retrieveOnlyNotesBeforeDate	Only retrieve notes created before this date. Can only be used in conjunction with f_retrieveOnlyNotesBeforeId.
long	f_retrieveOnlyNotesAfterId	Only retrieve notes with a later id. Can only be used in conjunction with f_retrieveOnlyNotesAfterDate.
long	f_retrieveOnlyNotesAfterDate	Only retrieve notes created after this date. Can only be used in conjunction with f_retrieveOnlyNotesAfterId.

## LIST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
Attachment[]	attachments	Contains all attachment resources.
TimelineUser	author	Contains the author of note.
date	creationDate	The date of creation of note.
TimelineNoteDiscussion	discussion	Contains information of the current discussion, where the note is parent or discussion part.
boolean	isDirectMessage	Indicates whether the note is a direct message or a normal message.

long	notedId	Is the identifier of an note.
integer	numberOfChildNotes	Number of all child notes.
integer	numberOfLikes	Number of users who like this note.
long	parentNotedId	Note identifier of the parent note. Is null when note is parent.
string[]	rights	Specifies the possibilities what the user can do with this note. <b>Constant Values</b> CAN_EDIT - Mark that the user can edit this note. CAN_REPLY - Mark that the user can reply this note. CAN_DELETE - Mark that the user can delete this note.
string	shortText	The short message of note.
TimelineNoteTag[]	tags	Tags of the note.
string	text	The message of the note.
TimelineTopic	topic	Contains the topic where the not was posted.
string[]	userNoteProperties	Specifies properties of the current user. Multiple markers can be set. <b>Constant Values</b> FAV - Mark for note is favor of current user. LIKE - Mark that the current user like this note. NOTIFY - Mark that the current user is notify in this note.
TimelineUser[]	usersToNotify	Contains all notified users.

#### 10.2.1.1 Property Resource

Resource for setting a property. It is a sub resource of a note.

#### Available HTTP-Methods for Property Resource

- [POST](#)
- [GET](#)
- [LIST](#)

---

#### POST - Property Resource

By sending a HTTP-POST request with all mandatory attributes, the server **creates** a new property resource and returns null.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking post requests on your server.*

#### POST - Url

.../timelineNotes/ID/propertys/

#### POST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	<b>key</b> ( <i>mandatory</i> )	Key of the property.
string	<b>keyGroup</b> ( <i>mandatory</i> )	KeyGroup of the property.
string	value	Value of property for specific key.

---

### GET - Property Resource

#### GET - Url

.../timelineNotes/ID/property/ID

#### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	<b>key</b> ( <i>mandatory</i> )	Key of the property.
string	<b>keyGroup</b> ( <i>mandatory</i> )	KeyGroup of the property.
long	propertyId	KeyGroup of the property.
string	value	Value of property for specific key.

---

### LIST - Property Resource

#### LIST - Url

.../timelineNotes/ID/property/

#### LIST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	key	Key of the property.
string	keyGroup	KeyGroup of the property.

#### 10.2.1.2 TimelineNoteDiscussion Resource

This resource represents an discussion of a timeline note. This resource conduce only for parsing of discussion in timeline note.

#### Available HTTP-Methods for TimelineNoteDiscussion Resource

- [GET](#)

---

### GET - TimelineNoteDiscussion Resource

#### GET - Url

.../timelineNotes/ID/timelineNoteDiscussions/ID

### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
integer	depth	This is the depth of the current note in discussion. Is only set if timeline notes filtered for discussionId
long	discussionId	Is the identifier of the current discussion.
integer	numberOfDiscussionNotes	Number of all notes in the discussion.

#### 10.2.1.3 TimelineNoteTag Resource

This resource represents an tag of a timeline note. This resource conduce only for parsing of discussion in timeline note.

### Available HTTP-Methods for TimelineNoteTag Resource

- [GET](#)
- [LIST](#)

---

### GET - TimelineNoteTag Resource

#### GET - Url

.../timelineNotes/ID/timelineNoteTags/ID

### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	name	This is the name of tag.
long	tagId	Is the identifier of tag.

---

### LIST - TimelineNoteTag Resource

#### LIST - Url

.../timelineNotes/ID/timelineNoteTags/

### LIST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	name	This is the name of tag.

long	tagId	Is the identifier of tag.
------	-------	---------------------------

# 11 TimelineTag Resource and Subresources

## 11.1 Overview

- [TimelineTag Resource](#)

## 11.2 Resources

### 11.2.1 TimelineTag Resource

This resource represents a tag of a note.

#### Available HTTP-Methods for TimelineTag Resource

- [POST](#)
- [LIST](#)

#### POST - TimelineTag Resource

##### POST - Url

.../timelineTags/

##### POST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
integer	offset	Specifies the offset of the list to return results for. <b>Default Value</b> is 0
integer	maxCount	Specifies the maximum amount of items in the resulting list. <b>Default Value</b> is 10. <b>Note:</b> Using it for retrieving the topic list the following parameter is required: <code>topicListType=LAST_MODIFIED!</code>
integer	checkAtLeastMoreResults	With this parameter it is possible to define a number of additional entities, which should be loaded when requesting data. These will be used to build information for paging, endless scrolling etc..
string[]	f_contentIds	Filters for notes with attachments with one of the given content identifiers. Since Communote version 3.3.
long	f_discussionId	Filters notes which are part of the specified discussion. If the discussion identifier is set the sort mode change from chronological (newest first) to threaded view style. Additional the discussion depth is set.



boolean	f_showFollowedItems	Filters notes which are followed by the user.
boolean	f_showNotesForMe	Filters notes which are addressed for the current user.
long[]	f_topicIds	Filters for notes with a topic being part of the given identifiers.
string[]	f_topicAliases	Filters for notes with topics being part of the given aliases.
long[]	f_topicIdsToExclude	Filters for notes NOT being part of the given topic identifiers.
long[]	f_tagIds	Filters for notes having all the tags assigned. The tag identifier refers to the tag database identifier.
string	f_tagPrefix	Filters for notes containing at least one tag starting with the given string.
long[]	f_userIds	Filters for notes with an author matching one of the given user identifiers.
string[]	f_userAliases	Filters for notes with an author matching one of the given user aliases.
date	f_startDate	Filters for notes being written after the specified date.
date	f_endDate	Filters for notes being written before or on the given date. <b>Note:</b> The specified day is included. If this day should be excluded, subtract 24 hours (86.400.000 ms) from f_endDate.
string	f_userSearchString	Filters for notes with an authors alias, last name or first name matching one of the given search string.
string	f_noteSearchString	Filters for notes containing the given string. <b>Note:</b> The results may vary depending on the Communote database used.
string	f_fullTextSearchString	Filters for notes where either the note content, the tags of the note or the name of the author contains the given string.
long	f_parentNoteId	Filters for notes with the specified parent's note id and the parent note itself.
string[]	f_repositoryIds	Filters for notes with attachments with the given repository identifiers. The order must match to the given f_contentIds parameter. Since Communote version 3.3.

boolean	f_showDirectMessages	Filters for notes which are direct messages.
boolean	f_showFavorites	Filters for notes which are marked as favorite.
boolean	f_includeComments	Specifies whether comments (or answers) of notes are included in the resulting list.
string	f_propertyFilter	Array of properties to filter for. The array is build like this: <i>FILTER_1, FILTER_2, ..., FILTER_3</i> where every FILTER looks like <code>&amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Example&amp;#34;,&amp;#34;value&amp;#34;:&amp;#34;[0:Property, 1:Group, 2:Key, 3:Value, 4:MatchMode, 5:Negate (Optional)]&amp;#125;,&amp;#34;whereas Property can be one of Note, Entity, Dummy and MatchMode can be one of EQUALS, STARTS_WITH, ENDS_WITH, CONTAINS, NOT_EQUALS. Negate is optional (default = false) can be true or false. <b>Note:</b> Every filter value has to be escaped with double quotes &amp;#34;. <b>Example:</b> <code>&amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Image&amp;#34;,&amp;#34;value&amp;#34;:&amp;#34;[&amp;#34;Note&amp;#34;,&amp;#34;com.communit&amp;#34;,&amp;#34;co</code></code>
string	f_topicAccess	Filter for notes which can be accessed at least by the given access level. <b>Constant Values</b> manager - Lists all topics of which the user is the manager. read - Lists all topics of which the user has read access. write - Lists all topics of which the user has write access.
boolean	f_showOnlyDiscussions	Show only the discussion.
boolean	f_showOnlyRootNotes	Show only root notes.
boolean	f_showDiscussionParticipation	Filter for discussions, the current user has participated in.
long	f_targetTopicId	Id of the topic to filter for.
long	f_noteId	Id of the note to filter for.
boolean	f_topicRelated	Specifies whether the results have to be topic/utp related.
long	f_retrieveOnlyNotesBeforeId	Only retrieve notes with a smaller id. Can only be used in conjunction with f_retrieveOnlyNotesBeforeDate.
long	f_retrieveOnlyNotesBeforeDate	Only retrieve notes created before this date. Can only be used in conjunction with f_retrieveOnlyNotesBeforeId.

long	f_retrieveOnlyNotesAfterId	Only retrieve notes with a later id. Can only be used in conjunction with f_retrieveOnlyNotesAfterDate.
long	f_retrieveOnlyNotesAfterDate	Only retrieve notes created after this date. Can only be used in conjunction with f_retrieveOnlyNotesAfterId.

## LIST - TimelineTag Resource

By sending a HTTP-GET request, the server **returns** a list of all requested timeline tag resources with the following attributes. The resulting list of tags will contain all tags of the last 1000 notes matching the filter parameters. The list can be used to show a tag cloud. Additionally there are two meta datas in the response. First with the number of all found elements and second whether there are more available elements.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking list requests on your server.*

### LIST - Url

.../timelineTags/

### LIST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
boolean	hideSelectedTags	If set, the selected tags will be excluded from the result.
string	tagCloudMode	Mode of the tag cloud. Must be one of 'PopularTags', 'MyTags' or 'SomeTags'
integer	numberOfLastDays	Parameter for the number of last days to consider (e.g. only retrieve the tags of the last 7 days).
integer	offset	Specifies the offset of the list to return results for. <b>Default Value</b> is 0
integer	maxCount	Specifies the maximum amount of items in the resulting list. <b>Default Value</b> is 10. <b>Note:</b> Using it for retrieving the topic list the following parameter is required: <code>topicListType=LAST_MODIFIED!</code>
integer	checkAtLeastMoreResults	With this parameter it is possible to define a number of additional entities, which should be loaded when requesting data. These will be used to build information for paging, endless scrolling etc..

string[]	f_contentIds	Filters for notes with attachments with one of the given content identifiers. Since Communote version 3.3.
long	f_discussionId	Filters notes which are part of the specified discussion. If the discussion identifier is set the sort mode change from chronological (newest first) to threaded view style. Additional the discussion depth is set.
boolean	f_showFollowedItems	Filters notes which are followed by the user.
boolean	f_showNotesForMe	Filters notes which are addressed for the current user.
long[]	f_topicIds	Filters for notes with a topic being part of the given identifiers.
string[]	f_topicAliases	Filters for notes with topics being part of the given aliases.
long[]	f_topicIdsToExclude	Filters for notes NOT being part of the given topic identifiers.
long[]	f_tagIds	Filters for notes having all the tags assigned. The tag identifier refers to the tag database identifier.
string	f_tagPrefix	Filters for notes containing at least one tag starting with the given string.
long[]	f_userIds	Filters for notes with an author matching one of the given user identifiers.
string[]	f_userAliases	Filters for notes with an author matching one of the given user aliases.
date	f_startDate	Filters for notes being written after the specified date.
date	f_endDate	Filters for notes being written before or on the given date. <b>Note:</b> The specified day is included. If this day should be excluded, subtract 24 hours (86.400.000 ms) from f_endDate.
string	f_userSearchString	Filters for notes with an authors alias, last name or first name matching one of the given search string.
string	f_noteSearchString	Filters for notes containing the given string. <b>Note:</b> The results may vary depending on the Communote database used.

string	f_fullTextSearchString	Filters for notes where either the note content, the tags of the note or the name of the author contains the given string.
long	f_parentNoteId	Filters for notes with the specified parent's note id and the parent note itself.
string[]	f_repositoryIds	Filters for notes with attachments with the given repository identifiers. The order must match to the given f_contentIds parameter. Since Communote version 3.3.
boolean	f_showDirectMessages	Filters for notes which are direct messages.
boolean	f_showFavorites	Filters for notes which are marked as favorite.
boolean	f_includeComments	Specifies whether comments (or answers) of notes are included in the resulting list.
string	f_propertyFilter	Array of properties to filter for. The array is build like this: <i>FILTER_1, FILTER_2, ..., FILTER_3</i> where every FILTER looks like <i>&amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Example&amp;#34;,&amp;#34;value&amp;#34;:&amp;#34;[0:Property, 1:Group, 2:Key, 3:Value, 4:MatchMode, 5:Negate (Optional)]&amp;#125;;</i> whereas <i>Property</i> can be one of <i>Note, Entity, Dummy</i> and <i>MatchMode</i> can be one of <i>EQUALS, STARTS_WITH, ENDS_WITH, CONTAINS, NOT_EQUALS</i> . <i>Negate</i> is optional (default = false) can be <i>true</i> or <i>false</i> . <b>Note:</b> Every filter value has to be escaped with double quotes <i>&amp;#34;</i> . <b>Example:</b> <i>&amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Image&amp;#34;,&amp;#34;value&amp;#34;:&amp;#34;Note&amp;#34;,&amp;#34;com.communote&amp;#34;,&amp;#34;co</i>
string	f_topicAccess	Filter for notes which can be accessed at least by the given access level. <b>Constant Values</b> <i>manager</i> - Lists all topics of which the user is the manager. <i>read</i> - Lists all topics of which the user has read access. <i>write</i> - Lists all topics of which the user has write access.
boolean	f_showOnlyDiscussions	Show only the discussion.
boolean	f_showOnlyRootNotes	Show only root notes.
boolean	f_showDiscussionParticipation	Filter for discussions, the current user has participated in.
long	f_targetTopicId	Id of the topic to filter for.

long	f_notelId	Id of the note to filter for.
boolean	f_topicRelated	Specifies whether the results have to be topic/utp related.
long	f_retrieveOnlyNotesBeforeId	Only retrieve notes with a smaller id. Can only be used in conjunction with f_retrieveOnlyNotesBeforeDate.
long	f_retrieveOnlyNotesBeforeDate	Only retrieve notes created before this date. Can only be used in conjunction with f_retrieveOnlyNotesBeforeId.
long	f_retrieveOnlyNotesAfterId	Only retrieve notes with a later id. Can only be used in conjunction with f_retrieveOnlyNotesAfterDate.
long	f_retrieveOnlyNotesAfterDate	Only retrieve notes created after this date. Can only be used in conjunction with f_retrieveOnlyNotesAfterId.

### LIST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	name	This is the name of the file.
integer	rank	This is the name of the file.
long	tagId	Is the Id of the tag.

## 12 TimelineTopic Resource and Subresources

### 12.1 Overview

- [TimelineTopic Resource](#)

### 12.2 Resources

#### 12.2.1 TimelineTopic Resource

This resource represents a topic of a note.

#### Available HTTP-Methods for TimelineTopic Resource

- [POST](#)
- [LIST](#)

#### POST - TimelineTopic Resource

#### POST - Url

.../timelineTopics/

#### POST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
integer	offset	Specifies the offset of the list to return results for. <b>Default Value</b> is 0
integer	maxCount	Specifies the maximum amount of items in the resulting list. <b>Default Value</b> is 10. <b>Note:</b> Using it for retrieving the topic list the following parameter is required: <code>topicListType=LAST_MODIFIED!</code>
integer	checkAtLeastMoreResults	With this parameter it is possible to define a number of additional entities, which should be loaded when requesting data. These will be used to build information for paging, endless scrolling etc..
string[]	f_contentIds	Filters for notes with attachments with one of the given content identifiers. Since Communote version 3.3.
long	f_discussionId	Filters notes which are part of the specified discussion. If the discussion identifier is set the sort mode change from chronological (newest first) to threaded view style. Additional the discussion depth is set.

boolean	f_showFollowedItems	Filters notes which are followed by the user.
boolean	f_showNotesForMe	Filters notes which are addressed for the current user.
long[]	f_topicIds	Filters for notes with a topic being part of the given identifiers.
string[]	f_topicAliases	Filters for notes with topics being part of the given aliases.
long[]	f_topicIdsToExclude	Filters for notes NOT being part of the given topic identifiers.
long[]	f_tagIds	Filters for notes having all the tags assigned. The tag identifier refers to the tag database identifier.
string	f_tagPrefix	Filters for notes containing at least one tag starting with the given string.
long[]	f_userIds	Filters for notes with an author matching one of the given user identifiers.
string[]	f_userAliases	Filters for notes with an author matching one of the given user aliases.
date	f_startDate	Filters for notes being written after the specified date.
date	f_endDate	Filters for notes being written before or on the given date. <b>Note:</b> The specified day is included. If this day should be excluded, subtract 24 hours (86.400.000 ms) from f_endDate.
string	f_userSearchString	Filters for notes with an authors alias, last name or first name matching one of the given search string.
string	f_noteSearchString	Filters for notes containing the given string. <b>Note:</b> The results may vary depending on the Communote database used.
string	f_fullTextSearchString	Filters for notes where either the note content, the tags of the note or the name of the author contains the given string.
long	f_parentNoteId	Filters for notes with the specified parent's note id and the parent note itself.
string[]	f_repositoryIds	Filters for notes with attachments with the given repository identifiers. The order must match to the given f_contentIds parameter. Since Communote version 3.3.



boolean	f_showDirectMessages	Filters for notes which are direct messages.
boolean	f_showFavorites	Filters for notes which are marked as favorite.
boolean	f_includeComments	Specifies whether comments (or answers) of notes are included in the resulting list.
string	f_propertyFilter	Array of properties to filter for. The array is build like this: <i>FILTER_1, FILTER_2, ..., FILTER_3</i> where every FILTER looks like <code>&amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Example&amp;#34;,&amp;#34;value&amp;#34;:&amp;#34;[0:Property, 1:Group, 2:Key, 3:Value, 4:MatchMode, 5:Negate (Optional)]&amp;#125;,&amp;#34;whereas Property can be one of Note, Entity, Dummy and MatchMode can be one of EQUALS, STARTS_WITH, ENDS_WITH, CONTAINS, NOT_EQUALS. Negate is optional (default = false) can be true or false. <b>Note:</b> Every filter value has to be escaped with double quotes &amp;#34;. <b>Example:</b> <code>&amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Image&amp;#34;,&amp;#34;value&amp;#34;:&amp;#34;[&amp;#34;Note&amp;#34;,&amp;#34;com.communit&amp;#34;,&amp;#34;co</code></code>
string	f_topicAccess	Filter for notes which can be accessed at least by the given access level. <b>Constant Values</b> manager - Lists all topics of which the user is the manager. read - Lists all topics of which the user has read access. write - Lists all topics of which the user has write access.
boolean	f_showOnlyDiscussions	Show only the discussion.
boolean	f_showOnlyRootNotes	Show only root notes.
boolean	f_showDiscussionParticipation	Filter for discussions, the current user has participated in.
long	f_targetTopicId	Id of the topic to filter for.
long	f_noteId	Id of the note to filter for.
boolean	f_topicRelated	Specifies whether the results have to be topic/utp related.
long	f_retrieveOnlyNotesBeforeId	Only retrieve notes with a smaller id. Can only be used in conjunction with f_retrieveOnlyNotesBeforeDate.
long	f_retrieveOnlyNotesBeforeDate	Only retrieve notes created before this date. Can only be used in conjunction with f_retrieveOnlyNotesBeforeId.

long	f_retrieveOnlyNotesAfterId	Only retrieve notes with a later id. Can only be used in conjunction with f_retrieveOnlyNotesAfterDate.
long	f_retrieveOnlyNotesAfterDate	Only retrieve notes created after this date. Can only be used in conjunction with f_retrieveOnlyNotesAfterId.

### LIST - TimelineTopic Resource

By sending a HTTP-GET request, the server **returns** a list of all requested timeline topic resources with the following attributes. The list will contain all topics which are part of the last 1000 notes matching the filter parameters. Additional there are two meta datas in the response. First with the number of all found elements and second whether there are more available elements.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking list requests on your server.*

### LIST - Url

.../timelineTopics/

### LIST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
integer	offset	Specifies the offset of the list to return results for. <b>Default Value</b> is 0
integer	maxCount	Specifies the maximum amount of items in the resulting list. <b>Default Value</b> is 10. <b>Note:</b> Using it for retrieving the topic list the following parameter is required: topicListType=LAST_MODIFIED!
integer	checkAtLeastMoreResults	With this parameter it is possible to define a number of additional entities, which should be loaded when requesting data. These will be used to build information for paging, endless scrolling etc..
string[]	f_contentIds	Filters for notes with attachments with one of the given content identifiers. Since Communote version 3.3.
long	f_discussionId	Filters notes which are part of the specified discussion. If the discussion identifier is set the sort mode change from chronological (newest first) to threaded view style. Additional the discussion depth is set.

boolean	f_showFollowedItems	Filters notes which are followed by the user.
boolean	f_showNotesForMe	Filters notes which are addressed for the current user.
long[]	f_topicIds	Filters for notes with a topic being part of the given identifiers.
string[]	f_topicAliases	Filters for notes with topics being part of the given aliases.
long[]	f_topicIdsToExclude	Filters for notes NOT being part of the given topic identifiers.
long[]	f_tagIds	Filters for notes having all the tags assigned. The tag identifier refers to the tag database identifier.
string	f_tagPrefix	Filters for notes containing at least one tag starting with the given string.
long[]	f_userIds	Filters for notes with an author matching one of the given user identifiers.
string[]	f_userAliases	Filters for notes with an author matching one of the given user aliases.
date	f_startDate	Filters for notes being written after the specified date.
date	f_endDate	Filters for notes being written before or on the given date. <b>Note:</b> The specified day is included. If this day should be excluded, subtract 24 hours (86.400.000 ms) from f_endDate.
string	f_userSearchString	Filters for notes with an authors alias, last name or first name matching one of the given search string.
string	f_noteSearchString	Filters for notes containing the given string. <b>Note:</b> The results may vary depending on the Communote database used.
string	f_fullTextSearchString	Filters for notes where either the note content, the tags of the note or the name of the author contains the given string.
long	f_parentNoteId	Filters for notes with the specified parent's note id and the parent note itself.
string[]	f_repositoryIds	Filters for notes with attachments with the given repository identifiers. The order must match to the given f_contentIds parameter. Since Communote version 3.3.

boolean	f_showDirectMessages	Filters for notes which are direct messages.
boolean	f_showFavorites	Filters for notes which are marked as favorite.
boolean	f_includeComments	Specifies whether comments (or answers) of notes are included in the resulting list.
string	f_propertyFilter	Array of properties to filter for. The array is build like this: <i>FILTER_1, FILTER_2, ..., FILTER_3</i> where every FILTER looks like <code>&amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Example&amp;#34;,&amp;#34;value&amp;#34;:&amp;#34;[0:Property, 1:Group, 2:Key, 3:Value, 4:MatchMode, 5:Negate (Optional)]&amp;#125;,&amp;#34;whereas Property can be one of Note, Entity, Dummy and MatchMode can be one of EQUALS, STARTS_WITH, ENDS_WITH, CONTAINS, NOT_EQUALS. Negate is optional (default = false) can be true or false. <b>Note:</b> Every filter value has to be escaped with double quotes &amp;#34;. <b>Example:</b> <code>&amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Image&amp;#34;,&amp;#34;value&amp;#34;:&amp;#34;[&amp;#34;Note&amp;#34;,&amp;#34;com.communit&amp;#34;,&amp;#34;co</code></code>
string	f_topicAccess	Filter for notes which can be accessed at least by the given access level. <b>Constant Values</b> manager - Lists all topics of which the user is the manager. read - Lists all topics of which the user has read access. write - Lists all topics of which the user has write access.
boolean	f_showOnlyDiscussions	Show only the discussion.
boolean	f_showOnlyRootNotes	Show only root notes.
boolean	f_showDiscussionParticipation	Filter for discussions, the current user has participated in.
long	f_targetTopicId	Id of the topic to filter for.
long	f_noteId	Id of the note to filter for.
boolean	f_topicRelated	Specifies whether the results have to be topic/utp related.
long	f_retrieveOnlyNotesBeforeId	Only retrieve notes with a smaller id. Can only be used in conjunction with f_retrieveOnlyNotesBeforeDate.
long	f_retrieveOnlyNotesBeforeDate	Only retrieve notes created before this date. Can only be used in conjunction with f_retrieveOnlyNotesBeforeId.

long	f_retrieveOnlyNotesAfterId	Only retrieve notes with a later id. Can only be used in conjunction with f_retrieveOnlyNotesAfterDate.
long	f_retrieveOnlyNotesAfterDate	Only retrieve notes created after this date. Can only be used in conjunction with f_retrieveOnlyNotesAfterId.

### LIST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	alias	This is the alias of the topic.
string	title	This is the title of the topic.
long	topicId	Is the Id of the topic.

## 13 TimelineUser Resource and Subresources

### 13.1 Overview

- [TimelineUser Resource](#)

### 13.2 Resources

#### 13.2.1 TimelineUser Resource

This resource represents a user of a note.

#### Available HTTP-Methods for TimelineUser Resource

- [POST](#)
- [LIST](#)

#### POST - TimelineUser Resource

#### POST - Url

.../timelineUsers/

#### POST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
integer	offset	Specifies the offset of the list to return results for. <b>Default Value</b> is 0
integer	maxCount	Specifies the maximum amount of items in the resulting list. <b>Default Value</b> is 10. <b>Note:</b> Using it for retrieving the topic list the following parameter is required: <code>topicListType=LAST_MODIFIED!</code>
integer	checkAtLeastMoreResults	With this parameter it is possible to define a number of additional entities, which should be loaded when requesting data. These will be used to build information for paging, endless scrolling etc..
string[]	f_contentIds	Filters for notes with attachments with one of the given content identifiers. Since Communote version 3.3.
long	f_discussionId	Filters notes which are part of the specified discussion. If the discussion identifier is set the sort mode change from chronological (newest first) to threaded view style. Additional the discussion depth is set.

boolean	f_showFollowedItems	Filters notes which are followed by the user.
boolean	f_showNotesForMe	Filters notes which are addressed for the current user.
long[]	f_topicIds	Filters for notes with a topic being part of the given identifiers.
string[]	f_topicAliases	Filters for notes with topics being part of the given aliases.
long[]	f_topicIdsToExclude	Filters for notes NOT being part of the given topic identifiers.
long[]	f_tagIds	Filters for notes having all the tags assigned. The tag identifier refers to the tag database identifier.
string	f_tagPrefix	Filters for notes containing at least one tag starting with the given string.
long[]	f_userIds	Filters for notes with an author matching one of the given user identifiers.
string[]	f_userAliases	Filters for notes with an author matching one of the given user aliases.
date	f_startDate	Filters for notes being written after the specified date.
date	f_endDate	Filters for notes being written before or on the given date. <b>Note:</b> The specified day is included. If this day should be excluded, subtract 24 hours (86.400.000 ms) from f_endDate.
string	f_userSearchString	Filters for notes with an authors alias, last name or first name matching one of the given search string.
string	f_noteSearchString	Filters for notes containing the given string. <b>Note:</b> The results may vary depending on the Communote database used.
string	f_fullTextSearchString	Filters for notes where either the note content, the tags of the note or the name of the author contains the given string.
long	f_parentNoteId	Filters for notes with the specified parent's note id and the parent note itself.
string[]	f_repositoryIds	Filters for notes with attachments with the given repository identifiers. The order must match to the given f_contentIds parameter. Since Communote version 3.3.

boolean	f_showDirectMessages	Filters for notes which are direct messages.
boolean	f_showFavorites	Filters for notes which are marked as favorite.
boolean	f_includeComments	Specifies whether comments (or answers) of notes are included in the resulting list.
string	f_propertyFilter	Array of properties to filter for. The array is build like this: <i>FILTER_1, FILTER_2, ..., FILTER_3</i> where every FILTER looks like <i>&amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Example&amp;#34;,&amp;#34;value&amp;#34;:&amp;#34;[0:Property, 1:Group, 2:Key, 3:Value, 4:MatchMode, 5:Negate (Optional)]&amp;#125;,&amp;#34;whereas Property can be one of Note, Entity, Dummy and MatchMode can be one of EQUALS, STARTS_WITH, ENDS_WITH, CONTAINS, NOT_EQUALS. Negate is optional (default = false) can be true or false. <b>Note:</b> Every filter value has to be escaped with double quotes &amp;#34;.&amp;#34; <b>Example:</b> &amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Image&amp;#34;,&amp;#34;value&amp;#34;:&amp;#34;[&amp;#34;Note&amp;#34;,&amp;#34;com.communit&amp;#34;,&amp;#34;co</i>
string	f_topicAccess	Filter for notes which can be accessed at least by the given access level. <b>Constant Values</b> <i>manager - Lists all topics of which the user is the manager.</i> <i>read - Lists all topics of which the user has read access.</i> <i>write - Lists all topics of which the user has write access.</i>
boolean	f_showOnlyDiscussions	Show only the discussion.
boolean	f_showOnlyRootNotes	Show only root notes.
boolean	f_showDiscussionParticipation	Filter for discussions, the current user has participated in.
long	f_targetTopicId	Id of the topic to filter for.
long	f_notId	Id of the note to filter for.
boolean	f_topicRelated	Specifies whether the results have to be topic/utp related.
long	f_retrieveOnlyNotesBeforeId	Only retrieve notes with a smaller id. Can only be used in conjunction with f_retrieveOnlyNotesBeforeDate.
long	f_retrieveOnlyNotesBeforeDate	Only retrieve notes created before this date. Can only be used in conjunction with f_retrieveOnlyNotesBeforeId.



long	f_retrieveOnlyNotesAfterId	Only retrieve notes with a later id. Can only be used in conjunction with f_retrieveOnlyNotesAfterDate.
long	f_retrieveOnlyNotesAfterDate	Only retrieve notes created after this date. Can only be used in conjunction with f_retrieveOnlyNotesAfterId.

## LIST - TimelineUser Resource

By sending a HTTP-GET request, the server **returns** a list of all requested timeline user resources with the following attributes. The list will contain all users which are authors of the last 1000 notes matching the filter parameters. Additional there are two meta datas in the response. First with the number of all found elements and second whether there are more available elements.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking list requests on your server.*

## LIST - Url

.../timelineUsers/

## LIST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
integer	offset	Specifies the offset of the list to return results for. <b>Default Value</b> is 0
integer	maxCount	Specifies the maximum amount of items in the resulting list. <b>Default Value</b> is 10. <b>Note:</b> Using it for retrieving the topic list the following parameter is required: topicListType=LAST_MODIFIED!
integer	checkAtLeastMoreResults	With this parameter it is possible to define a number of additional entities, which should be loaded when requesting data. These will be used to build information for paging, endless scrolling etc..
string[]	f_contentIds	Filters for notes with attachments with one of the given content identifiers. Since Communote version 3.3.
long	f_discussionId	Filters notes which are part of the specified discussion. If the discussion identifier is set the sort mode change from chronological (newest first) to threaded view style. Additional the discussion depth is set.

boolean	f_showFollowedItems	Filters notes which are followed by the user.
boolean	f_showNotesForMe	Filters notes which are addressed for the current user.
long[]	f_topicIds	Filters for notes with a topic being part of the given identifiers.
string[]	f_topicAliases	Filters for notes with topics being part of the given aliases.
long[]	f_topicIdsToExclude	Filters for notes NOT being part of the given topic identifiers.
long[]	f_tagIds	Filters for notes having all the tags assigned. The tag identifier refers to the tag database identifier.
string	f_tagPrefix	Filters for notes containing at least one tag starting with the given string.
long[]	f_userIds	Filters for notes with an author matching one of the given user identifiers.
string[]	f_userAliases	Filters for notes with an author matching one of the given user aliases.
date	f_startDate	Filters for notes being written after the specified date.
date	f_endDate	Filters for notes being written before or on the given date. <b>Note:</b> The specified day is included. If this day should be excluded, subtract 24 hours (86.400.000 ms) from f_endDate.
string	f_userSearchString	Filters for notes with an authors alias, last name or first name matching one of the given search string.
string	f_noteSearchString	Filters for notes containing the given string. <b>Note:</b> The results may vary depending on the Communote database used.
string	f_fullTextSearchString	Filters for notes where either the note content, the tags of the note or the name of the author contains the given string.
long	f_parentNoteId	Filters for notes with the specified parent's note id and the parent note itself.
string[]	f_repositoryIds	Filters for notes with attachments with the given repository identifiers. The order must match to the given f_contentIds parameter. Since Communote version 3.3.

boolean	f_showDirectMessages	Filters for notes which are direct messages.
boolean	f_showFavorites	Filters for notes which are marked as favorite.
boolean	f_includeComments	Specifies whether comments (or answers) of notes are included in the resulting list.
string	f_propertyFilter	Array of properties to filter for. The array is build like this: <i>FILTER_1, FILTER_2, ..., FILTER_3</i> where every FILTER looks like <code>&amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Example&amp;#34;,&amp;#34;value&amp;#34;:&amp;#34;[0:Property, 1:Group, 2:Key, 3:Value, 4:MatchMode, 5:Negate (Optional)]&amp;#125;,&amp;#34;whereas Property can be one of Note, Entity, Dummy and MatchMode can be one of EQUALS, STARTS_WITH, ENDS_WITH, CONTAINS, NOT_EQUALS. Negate is optional (default = false) can be true or false. <b>Note:</b> Every filter value has to be escaped with double quotes &amp;#34;. <b>Example:</b> <code>&amp;#123;&amp;#34;name&amp;#34;:&amp;#34;Image&amp;#34;,&amp;#34;value&amp;#34;:&amp;#34;[&amp;#34;Note&amp;#34;,&amp;#34;com.communit&amp;#34;,&amp;#34;co</code></code>
string	f_topicAccess	Filter for notes which can be accessed at least by the given access level. <b>Constant Values</b> manager - Lists all topics of which the user is the manager. read - Lists all topics of which the user has read access. write - Lists all topics of which the user has write access.
boolean	f_showOnlyDiscussions	Show only the discussion.
boolean	f_showOnlyRootNotes	Show only root notes.
boolean	f_showDiscussionParticipation	Filter for discussions, the current user has participated in.
long	f_targetTopicId	Id of the topic to filter for.
long	f_noteId	Id of the note to filter for.
boolean	f_topicRelated	Specifies whether the results have to be topic/utp related.
long	f_retrieveOnlyNotesBeforeId	Only retrieve notes with a smaller id. Can only be used in conjunction with f_retrieveOnlyNotesBeforeDate.
long	f_retrieveOnlyNotesBeforeDate	Only retrieve notes created before this date. Can only be used in conjunction with f_retrieveOnlyNotesBeforeId.

long	f_retrieveOnlyNotesAfterId	Only retrieve notes with a later id. Can only be used in conjunction with f_retrieveOnlyNotesAfterDate.
long	f_retrieveOnlyNotesAfterDate	Only retrieve notes created after this date. Can only be used in conjunction with f_retrieveOnlyNotesAfterId.

### LIST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	alias	This is the identifier name of the user. If user is permanently disabled or completely deleted alias is null. The "permanently disabled" status describes that user is anonymized and notes for user existing in database.
string	firstName	This is the firstname of the user.
string	lastName	This is the lastname of the user.
long	userId	Is the identifier of the user.

## 14 Topic Resource and Subresources

---

### 14.1 Overview

- [Topic Resource](#)
  - [ChildTopic Resource](#)
  - [ExternalObject Resource](#)
  - [Follow Resource](#)
  - [Property Resource](#)
  - [Right Resource](#)
  - [Role Resource](#)
  - [RoleBulkExternal Resource](#)
    - [RoleBulkExternalTopicRight Resource](#)
  - [RoleExternal Resource](#)

### 14.2 Resources

#### 14.2.1 Topic Resource

This resource represents a topic. Topics are used for categorising notes into different topics. Topics have a right management. That means that not everybody can read the notes of a topic. After creation of a topic the user can set the read and write rights to the topic.

#### Available HTTP-Methods for Topic Resource

- [POST](#)
- [PUT](#)
- [GET](#)
- [LIST](#)
- [DELETE](#)

---

#### POST - Topic Resource

By sending a HTTP-POST request with all mandatory attributes, the server **creates** a new topic resource and returns the topicId.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking post requests on your server.*

#### POST - Url

.../topics/

#### POST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
------	------	-------------

string	topicIdentifier	Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values. <b>Default Value</b> is "ID" <b>Constant Values</b> ID - <i>Topic Identifier is an Id</i> ALIAS - <i>Topic Identifier is an Alias</i> .
integer	offset	Specifies the offset of the list to return results for. <b>Default Value</b> is 0
integer	maxCount	Specifies the maximum amount of items in the resulting list. <b>Default Value</b> is 10. <b>Note:</b> Using it for retrieving the topic list the following parameter is required: <code>topicListType=LAST_MODIFIED!</code>
integer	checkAtLeastMoreResults	With this parameter it is possible to define a number of additional entities, which should be loaded when requesting data. These will be used to build information for paging, endless scrolling etc..

## POST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	alias	Is the alias to identify the topic. It is needed for the topic creation and can be used on a change request. <b>Default Value</b> is the formatted title.
boolean	allCanRead	Indicates whether everybody is allowed to read notes from this topic or not. <b>Default Value</b> is false.
boolean	allCanWrite	Indicates whether everybody is allowed to publish a note within this topic or not. <b>Default Value</b> is false.
string	description	What is this topic about.
ExternalObject[]	externalObjects	External objects of topic.
Property[]	properties	Properties of topic. Properties are key value pairs.
RoleExternal[]	roleExternals	External roles of topic.
Role[]	roles	Roles of topic.
Tag[]	tags	These tags are uses for searching with a tag cloud.

string	<b>title</b> ( <i>mandatory</i> )	Is the name of the topic.
--------	-----------------------------------	---------------------------

## PUT - Topic Resource

By sending a HTTP-PUT request with edited attributes, the server **edits** the topic resource and returns the topicId.

### PUT - Url

.../topics/ID?\_method=PUT

### PUT - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	topicIdentifier	Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values. <b>Default Value</b> is "ID" <b>Constant Values</b> ID - <i>Topic Identifier is an Id</i> ALIAS - <i>Topic Identifier is an Alias.</i>

## PUT - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	alias	Is the alias to identify the topic. It is needed for the topic creation and can be used on a change request. <b>Default Value</b> is the formatted title.
boolean	allCanRead	Indicates whether everybody is allowed to read notes from this topic or not. <b>Default Value</b> is false.
boolean	allCanWrite	Indicates whether everybody is allowed to publish a note within this topic or not. <b>Default Value</b> is false.
string	description	What is this topic about.
ExternalObject[]	externalObjects	External objects of topic.
Property[]	properties	Properties of topic. Properties are key value pairs.
RoleExternal[]	roleExternals	External roles of topic.
Role[]	roles	Roles of topic.
Tag[]	tags	These tags are uses for searching with a tag cloud.

string	title	Is the name of the topic.
string	<b>topicId</b> ( <i>mandatory</i> )	The key value to identify a topic.

### GET - Topic Resource

By sending a HTTP-GET request, the server **returns** the requested topic resource with the following attributes.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking get requests on your server.*

### GET - Url

.../topics/ID

### GET - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	topicIdentifier	Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values. <b>Default Value</b> is "ID" <b>Constant Values</b> ID - Topic Identifier is an Id ALIAS - Topic Identifier is an Alias.

### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	alias	Is the alias to identify the topic. It is needed for the topic creation and can be used on a change request. <b>Default Value</b> is the formatted title.
string	bannerImageId	Identifier of the banner image of the topic
string	bannerImageVersion	A string reflecting the version of the banner image. The string will change each time the banner image is updated. This value could for instance be used in the URL to the image to get the latest version from the server and not an outdated one from the client's cache.
ChildTopic[]	children	List of direct children



boolean	createSystemNotes	
string	description	What is this topic about.
boolean	isFollow	Indicates whether the topic is followed by the user.
date	lastBannerImageModificationDate	Deprecated: use bannerImageVersion instead. Timestamp of the last modification of the banner image. Can be used in the image URL to get the latest image and an old one from the client's cache.
date	lastModificationDate	Is used to specify the last modification date to list all topics that were modified afterwards. <b>Note:</b> Use only together with the topicListType=LAST_MODIFIED!
date	lastProfileImageModificationDate	Deprecated: use profileImageVersion instead. Timestamp of the last modification of the profile image. Can be used in the image URL to get the latest image and an old one from the client's cache.
string	profileImageId	Identifier of the profile image of the topic
string	profileImageVersion	A string reflecting the version of the profile image. The string will change each time the profile image is updated. This value could for instance be used in the URL to the image to get the latest version from the server and not an outdated one from the client's cache.
Tag[]	tags	These tags are used for searching with a tag cloud.
string	title	Is the name of the topic.
string	topicEmail	This is the mailing address of a topic, which enables the user to send regular e-mail to a topic
string	<b>topicId</b> (mandatory)	The key value to identify a topic.
string	userRole	Indicates the role of the user within this topic.

## LIST - Topic Resource

By sending a HTTP-GET request, the server **returns** a list of all requested topic resources with the following attributes. Additionally there are two meta data in the response. First with the number of all found elements and second whether there are more available elements.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking list requests on your server.*

**LIST - Url**

.../topics/

**LIST - Parameters**

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	topicListType	<p>Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values. <i>When topicListType is MOST_USED or LAST_USED there are no meta data in the response.</i></p> <p><b>Constant Values</b></p> <p>MANAGER - Lists all topics of which the user is the manager.</p> <p>READ - Lists all topics of which the user has the read access.</p> <p>WRITE - Lists all topics of which the user has the write access.</p> <p>MOST_USED - Lists all topics which are used mostly by the user.</p> <p>LAST_USED - Lists all topics chronical depending one the time of the last usage by the user.</p> <p>LAST_MODIFIED - Lists all topics chronical depending one the time of the last modification by the user.</p>
string[]	f_topicAliases	Filters topics which contain the specified aliases
long[]	f_topicIds	Filters topics which contain the specified identifier.
long[]	f_topicIdsToExclude	Filters topics whose identifier is specified to be excluded.
boolean	f_showFollowedItems	Filters topics which are followed.
string[]	f_tags	Filters topics which contain every single tag.
string	f_tagPrefix	Filters topics whose tags start with the specified string.
string	f_titleSearchString	Filters topics with string in title.
string	f_externalObjectSystemId	Filters topics which have an external object with the given system id. If f_externalObjectId is also set both must match the same external object. Only works for topicListType=READ, WRITE or MANAGE.

string	f_externalObjectId	Filters topics which have an external object with the given external id. If f_externalObjectSystemId is also set both must match the same external object. Only works for topicListType=READ, WRITE or MANAGE.
integer	offset	Specifies the offset of the list to return results for. <b>Default Value</b> is 0
integer	maxCount	Specifies the maximum amount of items in the resulting list. <b>Default Value</b> is 10. <b>Note:</b> Using it for retrieving the topic list the following parameter is required: topicListType=LAST_MODIFIED!
integer	checkAtLeastMoreResults	With this parameter it is possible to define a number of additional entities, which should be loaded when requesting data. These will be used to build information for paging, endless scrolling etc..

## LIST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	alias	Is the alias to identify the topic. It is needed for the topic creation and can be used on a change request. <b>Default Value</b> is the formatted title.
string	bannerImageId	Identifier of the banner image of the topic
string	bannerImageVersion	A string reflecting the version of the banner image. The string will change each time the banner image is updated. This value could for instance be used in the URL to the image to get the latest version from the server and not an outdated one from the client's cache.
ChildTopic[]	children	List of direct children
boolean	createSystemNotes	
string	description	What is this topic about.
boolean	isFollow	Indicates whether the topic is followed by the user.

date	lastBannerImageModificationDate	Deprecated: use bannerImageVersion instead. Timestamp of the last modification of the banner image. Can be used in the image URL to get the latest image and an old one from the client's cache.
date	lastModificationDate	Is used to specify the last modification date to list all topics that were modified afterwards. <b>Note:</b> Use only together with the topicListType=LAST_MODIFIED!
date	lastProfileImageModificationDate	Deprecated: use profileImageVersion instead. Timestamp of the last modification of the profile image. Can be used in the image URL to get the latest image and an old one from the client's cache.
string	profileImageId	Identifier of the profile image of the topic
string	profileImageVersion	A string reflecting the version of the profile image. The string will change each time the profile image is updated. This value could for instance be used in the URL to the image to get the latest version from the server and not an outdated one from the client's cache.
Tag[]	tags	These tags are used for searching with a tag cloud.
string	title	Is the name of the topic.
string	topicEmail	This is the mailing address of a topic, which enables the user to send regular e-mail to a topic
string	topicId	The key value to identify a topic.
string	userRole	Indicates the role of the user within this topic.

## DELETE - Topic Resource

By sending a HTTP-DELETE request, the server **deletes** the specified topic resource.

### DELETE - Url

.../topics/ID?\_method=DELETE

### DELETE - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
------	------	-------------

string	topicIdentifier	Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values. <b>Default Value</b> is "ID" <b>Constant Values</b> ID - <i>Topic Identifier is an Id</i> ALIAS - <i>Topic Identifier is an Alias.</i>
--------	-----------------	--

### DELETE - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	<b>topicId</b> ( <i>mandatory</i> )	The key value to identify a topic.

#### 14.2.1.1 ChildTopic Resource

This resource can be used to add or remove direct child topics for the given topic.

### Available HTTP-Methods for ChildTopic Resource

- [POST](#)
- [DELETE](#)

### POST - ChildTopic Resource

#### POST - Url

.../topics/ID/childTopics/

#### POST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	parentTopicIdentifier	Define the type the topics id is send, using one of the defined constant values. <b>Default Value</b> is "ID" <b>Constant Values</b> ID - <i>Topic Identifier is an Id</i> ALIAS - <i>Topic Identifier is an Alias.</i>

### POST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
long	<b>childTopicId</b> ( <i>mandatory</i> )	Id of the child topic.

---

**DELETE - ChildTopic Resource****DELETE - Url**

.../topics/ID/childTopics/ID?\_method=DELETE

**DELETE - Parameters**

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	parentTopicIdentifier	Define the type the topics id is send, using one of the defined constant values. <b>Default Value</b> is "ID" <b>Constant Values</b> ID - <i>Topic Identifier is an Id</i> ALIAS - <i>Topic Identifier is an Alias</i> .

**DELETE - Attributes**

Attributes are part of the entity send or requested.

Type	Name	Description
long	<b>childTopicId</b> ( <i>mandatory</i> )	Id of the child topic.

**14.2.1.2 ExternalObject Resource**

Resource to assign external object privileges to an internal topic.

**Available HTTP-Methods for ExternalObject Resource**

- [POST](#)
- [PUT](#)
- [GET](#)
- [LIST](#)
- [DELETE](#)

---

**POST - ExternalObject Resource****POST - Url**

.../topics/ID/externalObjects/

**POST - Parameters**

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
------	------	-------------

string	topicIdentifier	<p>Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values.</p> <p><b>Default Value</b> is "ID"</p> <p><b>Constant Values</b></p> <p>ID - <i>Topic Identifier is an Id</i></p> <p>ALIAS - <i>Topic Identifier is an Alias.</i></p>
--------	-----------------	---

### POST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	<b>externalId</b> ( <i>mandatory</i> )	Identifier of the external object in external system.
long	externalObjectId	Internal identifier of external object.
string	<b>externalSystemId</b> ( <i>mandatory</i> )	Identifier of the external system.
string	name	Name of external object.
string	topicAliasOfExternalObject	Is the alias to identify the topic. It is needed for the topic creation and can be used on a change request. Only set if topic independent filtering is used (topicId=0 in url).
long	topicIdOfExternalObject	The key value to identify a topic. Only set if topic independent filtering is used (topicId=0 in url).

---

### PUT - ExternalObject Resource

#### PUT - Url

.../topics/ID/externalObjects/ID?\_method=PUT

#### PUT - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	topicIdentifier	<p>Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values.</p> <p><b>Default Value</b> is "ID"</p> <p><b>Constant Values</b></p> <p>ID - <i>Topic Identifier is an Id</i></p> <p>ALIAS - <i>Topic Identifier is an Alias.</i></p>

**PUT - Attributes**

Attributes are part of the entity send or requested.

Type	Name	Description
string	externalId	Identifier of the external object in external system.
long	<b>externalObjectId</b> ( <i>mandatory</i> )	Internal identifier of external object.
string	externalSystemId	Identifier of the external system.
string	name	Name of external object.
string	topicAliasOfExternalObject	Is the alias to identify the topic. It is needed for the topic creation and can be used on a change request. Only set if topic independent filtering is used (topicId=0 in url).
long	topicIdOfExternalObject	The key value to identify a topic. Only set if topic independent filtering is used (topicId=0 in url).

**GET - ExternalObject Resource****GET - Url**

.../topics/ID/externalObjects/ID

**GET - Parameters**

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	topicIdentifier	Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values. <b>Default Value</b> is "ID" <b>Constant Values</b> ID - <i>Topic Identifier is an Id</i> ALIAS - <i>Topic Identifier is an Alias.</i>

**GET - Attributes**

Attributes are part of the entity send or requested.

Type	Name	Description
string	externalId	Identifier of the external object in external system.
long	<b>externalObjectId</b> ( <i>mandatory</i> )	Internal identifier of external object.
string	externalSystemId	Identifier of the external system.



string	name	Name of external object.
string	topicAliasOfExternalObject	Is the alias to identify the topic. It is needed for the topic creation and can be used on a change request. Only set if topic independent filtering is used (topicId=0 in url).
long	topicIdOfExternalObject	The key value to identify a topic. Only set if topic independent filtering is used (topicId=0 in url).

### LIST - ExternalObject Resource

By sending a HTTP-GET request, the server **returns** a list of all external object of the topic is returned. If topicId 0 is used and the parameters f\_externalSystemId and f\_externalObjectId are provided, ExternalObjects over all topics are considered (even if no access is allowed).

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking list requests on your server.*

### LIST - Url

.../topics/ID/externalObjects/

### LIST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	topicIdentifier	Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values. <b>Default Value</b> is "ID" <b>Constant Values</b> ID - <i>Topic Identifier is an Id</i> ALIAS - <i>Topic Identifier is an Alias.</i>

### LIST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	externalId	Identifier of the external object in external system.
long	externalObjectId	Internal identifier of external object.
string	externalSystemId	Identifier of the external system.
string	name	Name of external object.

string	topicAliasOfExternalObject	Is the alias to identify the topic. It is needed for the topic creation and can be used on a change request. Only set if topic independent filtering is used (topicId=0 in url).
long	topicIdOfExternalObject	The key value to identify a topic. Only set if topic independent filtering is used (topicId=0 in url).

---

## DELETE - ExternalObject Resource

### DELETE - Url

.../topics/ID/externalObjects/ID?\_method=DELETE

### DELETE - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	topicIdentifier	Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values. <b>Default Value</b> is "ID" <b>Constant Values</b> ID - <i>Topic Identifier is an Id</i> ALIAS - <i>Topic Identifier is an Alias</i> .

### DELETE - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	externalId	Identifier of the external object in external system.
long	externalObjectId	Internal identifier of external object.
string	externalSystemId	Identifier of the external system.

#### 14.2.1.3 Follow Resource

Resource for setting a follow. It is a sub resource of a topic.

### Available HTTP-Methods for Follow Resource

- [POST](#)
- [LIST](#)
- [DELETE](#)

---

## POST - Follow Resource

By sending a HTTP-POST request with all mandatory attributes, the server sets the follow status for specified user and returns null.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking post requests on your server.*

#### POST - Url

.../topics/ID/follows/

#### POST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
boolean	follow	follow status of the topic for the user. <b>Default Value</b> is false.

---

#### LIST - Follow Resource

##### LIST - Url

.../topics/ID/follows/

##### LIST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
boolean	follow	follow status of the topic for the user. <b>Default Value</b> is false.

---

#### DELETE - Follow Resource

##### DELETE - Url

.../topics/ID/follows/ID?\_method=DELETE

##### DELETE - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
------	------	-------------

#### 14.2.1.4 Property Resource

Resource for setting a property. It is a sub resource of a topic.

#### Available HTTP-Methods for Property Resource

- [POST](#)
  - [GET](#)
  - [LIST](#)
-

**POST - Property Resource**

By sending a HTTP-POST request with all mandatory attributes, the server **creates** a new property resource and returns null.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking post requests on your server.*

**POST - Url**

.../topics/ID/propertys/

**POST - Attributes**

Attributes are part of the entity send or requested.

Type	Name	Description
string	<b>key</b> (mandatory)	Key of the property.
string	<b>keyGroup</b> (mandatory)	KeyGroup of the property.
string	value	Value of property for specific key.

**GET - Property Resource****GET - Url**

.../topics/ID/propertys/ID

**GET - Attributes**

Attributes are part of the entity send or requested.

Type	Name	Description
string	<b>key</b> (mandatory)	Key of the property.
string	<b>keyGroup</b> (mandatory)	KeyGroup of the property.
long	propertyId	KeyGroup of the property.
string	value	Value of property for specific key.

**LIST - Property Resource****LIST - Url**

.../topics/ID/propertys/

**LIST - Attributes**

Attributes are part of the entity send or requested.

Type	Name	Description
string	key	Key of the property.
string	keyGroup	KeyGroup of the property.

### 14.2.1.5 Right Resource

Resource for setting a right. It is a sub resource of a topic.

#### Available HTTP-Methods for Right Resource

- [PUT](#)
- [GET](#)

---

#### PUT - Right Resource

By sending a HTTP-PUT request with edited attributes, the server **edits** the rights of topic and returns null.

#### PUT - Url

.../topics/ID/rights/ID?\_method=PUT

#### PUT - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
boolean	allCanRead	Indicates whether everybody is allowed to read notes from this topic or not. <b>Default Value</b> is false.
boolean	allCanWrite	Indicates whether everybody is allowed to publish a note within this topic or not. <b>Default Value</b> is false.
boolean	publicAccess	Specifies whether this topic is public accessibly. <b>Default Value</b> is false.

---

#### GET - Right Resource

#### GET - Url

.../topics/ID/rights/ID

#### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
boolean	allCanRead	Indicates whether everybody is allowed to read notes from this topic or not. <b>Default Value</b> is false.
boolean	allCanWrite	Indicates whether everybody is allowed to publish a note within this topic or not. <b>Default Value</b> is false.
long[]	managingUserIds	User identifier with managing access to topic.
boolean	publicAccess	Specifies whether this topic is public accessibly. <b>Default Value</b> is false.

long[]	readingUserIds	User identifier with reading access to topic.
long[]	writingUserIds	User identifier with writing access to topic.

#### 14.2.1.6 Role Resource

Resource for setting a topic role. It is a sub resource of a topic.

#### Available HTTP-Methods for Role Resource

- [POST](#)
- [PUT](#)
- [GET](#)
- [LIST](#)
- [DELETE](#)

#### POST - Role Resource

By sending a HTTP-POST request with all mandatory attributes, the server sets the role of topic for an user or group and returns null.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking post requests on your server.*

#### POST - Url

.../topics/ID/roles/

#### POST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	topicIdentifier	Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values. <b>Default Value</b> is "ID" <b>Constant Values</b> ID - <i>Topic Identifier is an Id</i> ALIAS - <i>Topic Identifier is an Alias.</i>

#### POST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
long	entityId	The identifier of the user or a group.
string	groupAlias	The alias of the group.

string	<b>role</b> ( <i>mandatory</i> )	Role for the user access to a topic. <b>Default Value</b> is "NONE". <b>Constant Values</b> NONE - A topic member with the NONE role has no access to the topic. MANAGER - A topic member with the MANAGER role has read, write and management access to a topic. MEMBER - A topic member with the MEMBER role has read and write access to a topic. VIEWER - A topic member with the VIEWER role only has read access to a topic.
string	userAlias	The alias of the user.

---

**PUT - Role Resource****PUT - Url**

.../topics/ID/roles/ID?\_method=PUT

**PUT - Parameters**

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	topicIdentifier	Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values. <b>Default Value</b> is "ID" <b>Constant Values</b> ID - Topic Identifier is an Id ALIAS - Topic Identifier is an Alias.

---

**GET - Role Resource****GET - Url**

.../topics/ID/roles/ID

**GET - Parameters**

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
------	------	-------------

string	topicIdentifier	<p>Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values.</p> <p><b>Default Value</b> is "ID"</p> <p><b>Constant Values</b></p> <p>ID - <i>Topic Identifier is an Id</i></p> <p>ALIAS - <i>Topic Identifier is an Alias.</i></p>
--------	-----------------	---

### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
------	------	-------------

### LIST - Role Resource

By sending a HTTP-GET request the server will return a list of all roles for the topic. Since Communote version 3.3.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking list requests on your server.*

### LIST - Url

.../topics/ID/roles/

### LIST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	topicIdentifier	<p>Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values.</p> <p><b>Default Value</b> is "ID"</p> <p><b>Constant Values</b></p> <p>ID - <i>Topic Identifier is an Id</i></p> <p>ALIAS - <i>Topic Identifier is an Alias.</i></p>

### LIST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	displayName	The display name for the user or group, includes both the full name and alias. Since Communote version 3.3.



long	entityId	The identifier of the user or a group.
string	groupAlias	The alias of the group.
boolean	isGroup	True if the entity is a group, false if it is a user. Since Communote version 3.3.
string	shortDisplayName	A shorter version of the display name for the user or group, includes only the name, or if not set only the alias. Since Communote version 3.3.
string	userAlias	The alias of the user.

---

### DELETE - Role Resource

#### DELETE - Url

.../topics/ID/roles/ID?\_method=DELETE

#### DELETE - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	topicIdentifier	Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values. <b>Default Value</b> is "ID" <b>Constant Values</b> ID - <i>Topic Identifier is an Id</i> ALIAS - <i>Topic Identifier is an Alias.</i>

#### 14.2.1.7 RoleBulkExternal Resource

Resource for setting a bulk of topic roles for an external system in combination with an external object. It is a sub resource of a topic.

#### Available HTTP-Methods for RoleBulkExternal Resource

- [POST](#)
- [PUT](#)
- [GET](#)
- [DELETE](#)

---

### POST - RoleBulkExternal Resource

By sending a HTTP-POST request with all mandatory attributes, the server sets roles of topic for an external system in combination with an external object and an external users or groups and returns null.

Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking post requests on your server.

### POST - Url

.../topics/ID/roleBulkExternals/

### POST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	topicIdentifier	Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values. <b>Default Value</b> is "ID" <b>Constant Values</b> ID - <i>Topic Identifier is an Id</i> ALIAS - <i>Topic Identifier is an Alias</i> .
RoleBulkExternalTopicRight[]	<b>topicRights</b> (mandatory)	rights for user or group as array with json objects
string	<b>externalSystemId</b> (mandatory)	Identifier for the external system (like "DefaultLDAP", "DefaultSharePoint", "DefaultConfluence").
string	<b>externalId</b> (mandatory)	The identifier of external object.

---

### PUT - RoleBulkExternal Resource

#### PUT - Url

.../topics/ID/roleBulkExternals/ID?\_method=PUT

#### PUT - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	topicIdentifier	Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values. <b>Default Value</b> is "ID" <b>Constant Values</b> ID - <i>Topic Identifier is an Id</i> ALIAS - <i>Topic Identifier is an Alias</i> .

---

### GET - RoleBulkExternal Resource

**GET - Url**

.../topics/ID/roleBulkExternals/ID

**GET - Parameters**

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	topicIdentifier	Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values. <b>Default Value</b> is "ID" <b>Constant Values</b> ID - <i>Topic Identifier is an Id</i> ALIAS - <i>Topic Identifier is an Alias.</i>

**GET - Attributes**

Attributes are part of the entity send or requested.

Type	Name	Description
------	------	-------------

**DELETE - RoleBulkExternal Resource****DELETE - Url**

.../topics/ID/roleBulkExternals/ID?\_method=DELETE

**DELETE - Parameters**

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	topicIdentifier	Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values. <b>Default Value</b> is "ID" <b>Constant Values</b> ID - <i>Topic Identifier is an Id</i> ALIAS - <i>Topic Identifier is an Alias.</i>

**14. RoleBulkExternalTopicRight Resource**

Resource for setting a bulk of topic roles for an external system. It is a sub resource of a RoleBulkExternal and is used only as container for serial- and deserialization.

**Available HTTP-Methods for RoleBulkExternalTopicRight Resource**

- [GET](#)

---

**GET - RoleBulkExternalTopicRight Resource****GET - Url**

.../topics/ID/roleBulkExternals/ID/roleBulkExternalTopicRights/ID

**GET - Attributes**

Attributes are part of the entity send or requested.

Type	Name	Description
string	externalGroupId	rights for user or group as array with json objects
string	externalUserId	rights for user or group as array with json objects
string	<b>role</b> ( <i>mandatory</i> )	<p>Role for the user access to a topic.</p> <p><b>Constant Values</b></p> <p>MANAGER - A topic member with the MANAGER role has read, write and management access to a topic.</p> <p>MEMBER - A topic member with the MEMBER role has read and write access to a topic.</p> <p>VIEWER - A topic member with the VIEWER role only has read access to a topic.</p>

**14.2.1.8 RoleExternal Resource**

Resource for setting a topic role for an external system in combination with an external object. It is a sub resource of a topic. **Default Value** is "NONE".

**Available HTTP-Methods for RoleExternal Resource**

- [POST](#)
- [PUT](#)
- [GET](#)
- [LIST](#)
- [DELETE](#)

---

**POST - RoleExternal Resource**

By sending a HTTP-POST request with all mandatory attributes, the server sets the role of topic for an external system in combination with an external object and an external user or group and returns null.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking post requests on your server.*

**POST - Url**

.../topics/ID/roleExternals/

**POST - Parameters**

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	topicIdentifier	Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values. <b>Default Value</b> is "ID" <b>Constant Values</b> ID - <i>Topic Identifier is an Id</i> ALIAS - <i>Topic Identifier is an Alias.</i>

### POST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	externalGroupId	The identifier of the external group.
string	<b>externalId</b> ( <i>mandatory</i> )	The identifier of external object.
string	<b>externalSystemId</b> ( <i>mandatory</i> )	The external system identifier.
string	externalUserId	The identifier of the external user.
string	<b>role</b> ( <i>mandatory</i> )	Role for the user access to a topic. <b>Constant Values</b> NONE - <i>A topic member with the NONE role has no access to the topic.</i> MANAGER - <i>A topic member with the MANAGER role has read, write and management access to a topic.</i> MEMBER - <i>A topic member with the MEMBER role has read and write access to a topic.</i> VIEWER - <i>A topic member with the VIEWER role only has read access to a topic.</i>

### PUT - RoleExternal Resource

#### PUT - Url

.../topics/ID/roleExternals/ID?\_method=PUT

#### PUT - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
------	------	-------------

string	topicIdentifier	<p>Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values.</p> <p><b>Default Value</b> is "ID"</p> <p><b>Constant Values</b></p> <p>ID - <i>Topic Identifier is an Id</i></p> <p>ALIAS - <i>Topic Identifier is an Alias.</i></p>
--------	-----------------	---

## GET - RoleExternal Resource

### GET - Url

.../topics/ID/roleExternals/ID

### GET - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	topicIdentifier	<p>Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values.</p> <p><b>Default Value</b> is "ID"</p> <p><b>Constant Values</b></p> <p>ID - <i>Topic Identifier is an Id</i></p> <p>ALIAS - <i>Topic Identifier is an Alias.</i></p>

### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
------	------	-------------

## LIST - RoleExternal Resource

### LIST - Url

.../topics/ID/roleExternals/

### LIST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
------	------	-------------

string	topicIdentifier	<p>Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values.</p> <p><b>Default Value</b> is "ID"</p> <p><b>Constant Values</b></p> <p>ID - <i>Topic Identifier is an Id</i></p> <p>ALIAS - <i>Topic Identifier is an Alias.</i></p>
--------	-----------------	---

## DELETE - RoleExternal Resource

### DELETE - Url

.../topics/ID/roleExternals/ID?\_method=DELETE

### DELETE - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	topicIdentifier	<p>Topic lists have different views. These view can result from rights issues or usage issues. This requires a value to specify the type of list you want to request, using the defined constant values.</p> <p><b>Default Value</b> is "ID"</p> <p><b>Constant Values</b></p> <p>ID - <i>Topic Identifier is an Id</i></p> <p>ALIAS - <i>Topic Identifier is an Alias.</i></p>

## 15 User Resource and Subresources

---

### 15.1 Overview

- [User Resource](#)
  - [ExternalLogin Resource](#)
    - [ExternalProperty Resource](#)
  - [Follow Resource](#)
  - [Image Resource](#)
  - [NavigationItem Resource](#)
  - [Observation Resource](#)
  - [Property Resource](#)
  - [UserPreference Resource](#)
    - [Preference Resource](#)

### 15.2 Resources

#### 15.2.1 User Resource

This is the resource to define the users within the system.

##### Available HTTP-Methods for User Resource

- [POST](#)
- [PUT](#)
- [GET](#)
- [LIST](#)

---

#### POST - User Resource

##### POST - Url

.../users/

##### POST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
integer	offset	Specifies the offset of the list to return results for. <b>Default Value</b> is 0
integer	maxCount	Specifies the maximum amount of items in the resulting list. <b>Default Value</b> is 10. <b>Note:</b> Using it for retrieving the topic list the following parameter is required: <code>topicListType=LAST_MODIFIED!</code>



integer	checkAtLeastMoreResults	With this parameter it is possible to define a number of additional entities, which should be loaded when requesting data. These will be used to build information for paging, endless scrolling etc..
---------	-------------------------	--

### POST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
Property[]	properties	Properties of user. Properties are key value pairs.

---

### PUT - User Resource

By sending a HTTP-PUT request with edited attributes, the server **edits** the user resource and returns the userId.

#### PUT - Url

.../users/ID?\_method=PUT

#### PUT - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	firstName	This is the firstname of the user.
string	lastName	This is the surname of the user.
Property[]	properties	Properties of user. Properties are key value pairs.
Tag[]	tags	Indicates whether the tags with witch the user is tagged.
long	userId (mandatory)	The identifier of the user.

---

### GET - User Resource

By sending a HTTP-GET request, the server **returns** the requested user resource with the following attributes.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking get requests on your server.*

#### GET - Url

.../users/ID

#### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	alias	This is the alias of the user. It is also used for authentication.
string	company	The users company.
string	faxAreaCode	Area code of the user fax number.
string	faxCountryCode	Country code of the user fax number.
string	faxNumber	Number of the user fax number.
string	firstName	This is the firstname of the user.
boolean	isFollow	Indicates whether the author is followed by the user.
string	languageLocale	This is the locale of the user.
date	lastModificationDate	This is the date on which the user change some personal profil information.
string	lastName	This is the surname of the user.
date	lastPhotoModificationDate	Deprecated: use profileImageVersion instead. The date of the last modification of the user profile image.
string	phoneAreaCode	Area code of the user phone number.
string	phoneCountryCode	Country code of the user phone number.
string	phoneNumber	Number of the user phone number.
string	position	The users position.
string	profileImageVersion	A string reflecting the version of the profile image. The string will change when the profile image is updated or another image provider is used. This value could for instance be used in the URL to the image to get the latest version from the server and not an outdated one from the client's cache.
string	salutation	The salutation of the user will be used within notification messages.
Tag[]	tags	Indicates whether the tags with witch the user is tagged.
string	timeZoneld	This is the identifier of the time zone where the user is.
long	<b>userId</b> ( <i>mandatory</i> )	The identifier of the user.

---

## LIST - User Resource

By sending a HTTP-GET request, the server **returns** a list of all requested user resources with the following attributes. Additionally there are two meta datas in the response. First with the number of all found elements and second whether there are more available elements.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking list requests on your server.*

### LIST - Url

.../users/

### LIST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	searchString	Searching users information for a substring. Searching in alias, lastname and firstname.
string	f_userAlias	Get user information by alias. When get user by alias is set, there are no meta datas in the response
integer	offset	Specifies the offset of the list to return results for. <b>Default Value</b> is 0
integer	maxCount	Specifies the maximum amount of items in the resulting list. <b>Default Value</b> is 10. <b>Note:</b> Using it for retrieving the topic list the following parameter is required: <code>topicListType=LAST_MODIFIED!</code>
integer	checkAtLeastMoreResults	With this parameter it is possible to define a number of additional entities, which should be loaded when requesting data. These will be used to build information for paging, endless scrolling etc..

### LIST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	alias	This is the alias of the user. It is also used for authentication.
string	company	The users company.
string	faxAreaCode	Area code of the user fax number.
string	faxCountryCode	Country code of the user fax number.
string	faxNumber	Number of the user fax number.
string	firstName	This is the firstname of the user.

boolean	isFollow	Indicates whether the author is followed by the user.
string	languageLocale	This is the locale of the user.
date	lastModificationDate	This is the date on which the user change some personal profil information.
string	lastName	This is the surname of the user.
date	lastPhotoModificationDate	Deprecated: use profileImageVersion instead. The date of the last modification of the user profile image.
string	phoneAreaCode	Area code of the user phone number.
string	phoneCountryCode	Country code of the user phone number.
string	phoneNumber	Number of the user phone number.
string	position	The users position.
string	profileImageVersion	A string reflecting the version of the profile image. The string will change when the profile image is updated or another image provider is used. This value could for instance be used in the URL to the image to get the latest version from the server and not an outdated one from the client's cache.
string	salutation	The salutation of the user will be used within notification messages.
Tag[]	tags	Indicates whether the tags with witch the user is tagged.
string	timeZoneId	This is the identifier of the time zone where the user is.
long	userId	The identifier of the user.

#### 15.2.1.1 ExternalLogin Resource

This resource can be used to handle the external logins. Since Communote version 3.3.

#### Available HTTP-Methods for ExternalLogin Resource

- [LIST](#)

---

#### LIST - ExternalLogin Resource

##### LIST - Url

.../users/ID/externalLogins/

##### LIST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
long	databaseId	Backend database id
string	externalLoginId	The login id for the user in the external system.
string	externalSystemId	The identifier of the external system this login belongs to, e.g. DefaultLDAP, DefaultConfluence
string	permanentId	The permanent identifier of the external login.
ExternalProperty[]	properties	Properties for the external login.

### 15. ExternalProperty Resource

Resource for setting a property. It is a sub resource of a external login, cannot be set, only returned with ExternalLogin Resource. Since Communote version 3.3.

#### Available HTTP-Methods for ExternalProperty Resource

- [GET](#)

---

#### GET - ExternalProperty Resource

##### GET - Url

.../users/ID/externalLogins/ID/externalProperty/ID

##### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	key	Key of the property.
string	keyGroup	KeyGroup of the property.
long	propertyId	Dummy Id of the property.
string	value	Value of property for specific key.

### 15.2.1.2 Follow Resource

Resource for setting a follow. It is a sub resource of a user.

#### Available HTTP-Methods for Follow Resource

- [POST](#)
- [LIST](#)
- [DELETE](#)

---

#### POST - Follow Resource

By sending a HTTP-POST request with all mandatory attributes, the server sets the follow status for specified user and returns null.

Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking post requests on your server.

### POST - Url

.../users/ID/follows/

### POST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
boolean	follow	follow status of the user for the current user. <b>Default Value</b> is false.

---

### LIST - Follow Resource

### LIST - Url

.../users/ID/follows/

### LIST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
boolean	follow	follow status of the user for the current user. <b>Default Value</b> is false.

---

### DELETE - Follow Resource

### DELETE - Url

.../users/ID/follows/ID?\_method=DELETE

### DELETE - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
------	------	-------------

#### 15.2.1.3 Image Resource

This is an image resource. It is a sub resource of a user.

### Available HTTP-Methods for Image Resource

- [GET](#)

---

### GET - Image Resource

By sending a HTTP-GET request, the server **returns** a stream containing the image as binary.

Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking get requests on your server.

### GET - Url

.../users/ID/images/ID

### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	<b>size</b> (mandatory)	This is the size of of the image. <b>Constant Values</b> SMALL - Returns an image of small size (25x25). MEDIUM - Returns an image of medium size (50x50). LARGE - Returns an image of large size (100x100).

#### 15.2.1.4 NavigationItem Resource

This resource can be used to handle navigation items.

#### Available HTTP-Methods for NavigationItem Resource

- [POST](#)
- [PUT](#)
- [GET](#)
- [LIST](#)
- [DELETE](#)

---

### POST - NavigationItem Resource

#### POST - Url

.../users/ID/navigationItems/

#### POST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	data	Payload of the item. Note: this is a JSON object formatted as string
integer	index	Index of the item
long	lastAccessDate	The last access date as long.
string	name	Name of the item

---

### PUT - NavigationItem Resource

#### PUT - Url

.../users/ID/navigationItems/ID?\_method=PUT

### PUT - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	data	Payload of the item. Note: this is a JSON object formatted as string
integer	index	Index of the item
long	lastAccessDate	The last access date as long.
string	name	Name of the item
long	navigationItemId	Id of the navigation item

---

### GET - NavigationItem Resource

#### GET - Url

.../users/ID/navigationItems/ID

#### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
long	navigationItemId	Id of the navigation item

---

### LIST - NavigationItem Resource

#### LIST - Url

.../users/ID/navigationItems/

#### LIST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
long[]	f_navigationItemIds	Observations to get results for, If null or empty all observations will be checked.

---

### DELETE - NavigationItem Resource

#### DELETE - Url

.../users/ID/navigationItems/ID?\_method=DELETE

#### DELETE - Attributes

Attributes are part of the entity send or requested.



Type	Name	Description
long	navigationItemId	Id of the navigation item

#### 15.2.1.5 Observation Resource

This resource can be used to check the state if the users observations.

#### Available HTTP-Methods for Observation Resource

- [GET](#)
- [LIST](#)

---

#### GET - Observation Resource

##### GET - Url

.../users/ID/observations/ID

##### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
long	checkDate	Date the check was done.
long	count	Number of notes written since the last check date for this observation
long	observationId	Id of the observation

---

#### LIST - Observation Resource

##### LIST - Url

.../users/ID/observations/

##### LIST - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
long[]	f_observations	Observations to get results for, If null or empty all observations will be checked.
long	f_lastCheckDate	Date the last check was done.

#### 15.2.1.6 Property Resource

Resource for setting a property. It is a sub resource of a user.

#### Available HTTP-Methods for Property Resource

- [POST](#)
- [GET](#)

- [LIST](#)

---

### POST - Property Resource

By sending a HTTP-POST request with all mandatory attributes, the server **creates** a new property resource and returns null.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking post requests on your server.*

#### POST - Url

.../users/ID/propertys/

#### POST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	<b>key</b> (mandatory)	Key of the property.
string	<b>keyGroup</b> (mandatory)	KeyGroup of the property.
string	value	Value of property for specific key.

---

### GET - Property Resource

#### GET - Url

.../users/ID/propertys/ID

#### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	<b>key</b> (mandatory)	Key of the property.
string	<b>keyGroup</b> (mandatory)	KeyGroup of the property.
long	propertyId	KeyGroup of the property.
string	value	Value of property for specific key.

---

### LIST - Property Resource

#### LIST - Url

.../users/ID/propertys/

#### LIST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	key	Key of the property.

string	keyGroup	KeyGroup of the property.
--------	----------	---------------------------

#### 15.2.1.7 UserPreference Resource

Resource for retrieving and setting user preferences. Whatever user id or preference id you use, this will always return results for the current user.

#### Available HTTP-Methods for UserPreference Resource

- [PUT](#)
- [GET](#)

---

#### PUT - UserPreference Resource

##### PUT - Url

.../users/ID/userPreferences/ID?\_method=PUT

##### PUT - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	<b>className</b> ( <i>mandatory</i> )	Name of the implementing class.
long	preferenceId	Id of the preference, this can be ignored as it is ignored from backend.
Preference[]	preferences	The values of the preferences

---

#### GET - UserPreference Resource

Method to get values of a specific preference.

*Please notice the documentation for [Resources Tunneling](#), if you encounter any problems invoking get requests on your server.*

##### GET - Url

.../users/ID/userPreferences/ID

##### GET - Parameters

Parameters are generally used to filter the result and are not part of the result set.

Type	Name	Description
string	f_className	Filter for this preferences.

##### GET - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	className	Name of the implementing class.

long	preferenceId	Id of the preference, this can be ignored as it is ignored from backend.
Preference[]	preferences	The values of the preferences

### 15. Preference Resource

Resource for preference maps. It is a sub resource of Preferences, but can't be called directly.

#### Available HTTP-Methods for Preference Resource

- [POST](#)
- [PUT](#)
- [GET](#)
- [DELETE](#)

---

#### POST - Preference Resource

##### POST - Url

.../users/ID/userPreferences/ID/preferences/

##### POST - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	key	Key of the preference.
string	value	Value of preference for specific key.

---

#### PUT - Preference Resource

##### PUT - Url

.../users/ID/userPreferences/ID/preferences/ID?\_method=PUT

##### PUT - Attributes

Attributes are part of the entity send or requested.

Type	Name	Description
string	key	Key of the preference.
long	<b>preferenceEntryId</b> ( <i>mandatory</i> )	Id of the preference, this can be ignored.
string	value	Value of preference for specific key.

---

#### GET - Preference Resource

##### GET - Url

.../users/ID/userPreferences/ID/preferences/ID

**GET - Attributes**

Attributes are part of the entity send or requested.

Type	Name	Description
long	<b>preferenceEntryId</b> ( <i>mandatory</i> )	Id of the preference, this can be ignored.

---

**DELETE - Preference Resource****DELETE - Url**

.../users/ID/userPreferences/ID/preferences/ID?\_method=DELETE

**DELETE - Attributes**

Attributes are part of the entity send or requested.

Type	Name	Description
long	<b>preferenceEntryId</b> ( <i>mandatory</i> )	Id of the preference, this can be ignored.

## 16 Usage of Communote Client for Java

The Communote Client for Java provides functionality to authenticate to communote and to access resources. Furthermore he contains java representations of resources.

Here are some important informations to use the Communote Client as an dependency in applications.

*Please note that the Communote Client for Java has only development status! So we not guarantee for entire functionality to access the Communote REST API.*

If you want to use the Communote Client following constructor parameter are necessary:

Type	Name	Description
String	baseURL	URL of API of system you want to access (http://.../microblog/global/api/rest/v3_0)
String	username	login name of user who wants to access
String	password	password of user

An overview of all Client methods you can find [below](#).

### 16.1 Important Classes

Every resource has his own package. For example the topic resource is located in that package:

```
com.communote.plugins.api.rest.client.resource.topic;
```

Subresources are located in subpackages and resources of subresources in subsubpackages an so on.

In every resource package you can found following classes.

#### 16.1.1 Resource Classes

Resource classes represents the result of an request. So every resource class is an deserialized form of the returned GET request.

**For an specific information about the attributes of an returned resource read the attribute documentation part of appropriate resource for GET and LIST ( [Resource Documentation](#)).**

#### 16.1.2 Parameter Classes

Every parameter classes represent an HTTP-Request to create, get, edit, list or delete the appropriate resource.

For the naming of the parameter class the Client used following convention:

HTTP-Method	Name of Parameter Class	Example for Topic
POST	Create...Parameter	CreateTopicParameter
PUT	Edit...Parameter	EditTopicParameter
GET	Get...Parameter	GetTopicParameter
LIST	Get...ResourceParameter	GetCollectionTopicParameter

DELETE	Delete...Parameter	DeleteTopicParameter
--------	--------------------	----------------------

*Take care that you choose the right parameter class. Some parameter classes have the same name so you must be careful that you have the right class. You can see this in the package path of the parameter class.*

```
com.communote.plugins.api.rest.client.resource.topic.property;
com.communote.plugins.api.rest.client.resource.note.property;
```

The parameter classes contains all necessary parameters and attributes for the request with the appropriate HTTP-Method. Besides this, every parameter class contain tree important methods:

Method	Return	Description
getUriPath()	String	Get the path to resource
isValidParameterList()	Boolean	Valid that all manatory parameters / attributes are set
getResourceJSONRepresentation()	String	Get the serialize parameter class as json

**Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.**

For example you want to know witch parameters or attributes are necessary to create an topic with `CreateTopicParameter` or you need to know how to use them?

So check the reference of topic resource with an HTTP POST ( [convention table above](#)) request.

### 16.1.3 Enumeration for Constants

All java files with an "E" at the beginning and after this any other upper case letter are enumerations. Enumerations are used for constant values of attributes and parameters.

## 16.2 Overview of Communote Client Resource Methods

### 16.2.1 Attachment

#### Methods

*Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.*

Name	Parameter	Return
createAttachment	CreateAttachmentParameter	Long
getAttachment	GetAttachmentParameter	AttachmentResource
deleteAttachment	DeleteAttachmentParameter	void

#### Package

*Informations about packages and contained classes in packages are listed unter [Important Classes](#).*

com.communote.plugins.api.rest.client.resource.attachment

### Package - Files

- AttachmentResource.java
- CreateAttachmentParameter.java
- GetAttachmentParameter.java
- DeleteAttachmentParameter.java

#### 16.2.1.1 Preview

### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
getAttachmentPreview	GetPreviewParameter	PreviewResource

### Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

com.communote.plugins.api.rest.client.resource.attachment.preview

### Package - Files

- PreviewResource.java
- GetPreviewParameter.java

#### 16.2.2 Group

### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
getGroup	GetGroupParameter	GroupResource

### Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

com.communote.plugins.api.rest.client.resource.group

### Package - Files

- GroupResource.java
- GetGroupParameter.java
- EGroupIdentifier.java

#### 16.2.2.1 Member

### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.



Name	Parameter	Return
getCollectionGroupMember	GetCollectionMemberParameter	List<MemberResource>

### Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.group.member`

### Package - Files

- MemberResource.java
- GetCollectionMemberParameter.java
- EGroupIdentifier.java

### 16.2.3 LastModificationDate

#### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributes and parameters in every parameter class.

Name	Parameter	Return
getCollectionLastModificationDate	GetCollectionLastModificationDateParameter	List<LastModificationDateResource>

### Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.lastmodificationdate`

### Package - Files

- LastModificationDateResource.java
- GetCollectionLastModificationDateParameter.java
- EType.java

### 16.2.4 Note

#### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributes and parameters in every parameter class.

Name	Parameter	Return
createNote	CreateNoteParameter	Long
editNote	EditNoteParameter	Long
getNote	GetNoteParameter	NoteResource
deleteNote	DeleteNoteParameter	void

### Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

com.communote.plugins.api.rest.client.resource.note

### Package - Files

- NoteResource.java
- CreateNoteParameter.java
- EditNoteParameter.java
- GetNoteParameter.java
- DeleteNoteParameter.java

#### 16.2.4.1 Favorite

### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
createFavorite	CreateFavoriteParameter	Long
getCollectionNoteFavorite	GetCollectionFavoriteParameter	List<FavoriteResource>
deleteFavorite	DeleteFavoriteParameter	void

### Package

Informations about packages and contained classes in packages are listed unter [Important Classes](#).

com.communote.plugins.api.rest.client.resource.note.favorite

### Package - Files

- FavoriteResource.java
- CreateFavoriteParameter.java
- GetCollectionFavoriteParameter.java
- DeleteFavoriteParameter.java

#### 16.2.4.2 Like

### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
createLike	CreateLikeParameter	Long
getCollectionNoteLike	GetCollectionLikeParameter	List<LikeResource>
deleteLike	DeleteLikeParameter	void

### Package

Informations about packages and contained classes in packages are listed unter [Important Classes](#).

com.communote.plugins.api.rest.client.resource.note.like

### Package - Files

- LikeResource.java

- CreateLikeParameter.java
- GetCollectionLikeParameter.java
- DeleteLikeParameter.java

#### 16.2.4.3 Property

##### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
createProperty	CreatePropertyParameter	Long
getNoteProperty	GetPropertyParameter	PropertyResource
getCollectionNoteProperty	GetCollectionPropertyParameter	List<PropertyResource>

##### Package

Informations about packages and contained classes in packages are listed unter [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.note.property`

##### Package - Files

- PropertyResource.java
- CreatePropertyParameter.java
- GetPropertyParameter.java
- GetCollectionPropertyParameter.java

#### 16.2.5 Tag

##### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
editTag	EditTagParameter	Long
getTag	GetTagParameter	TagResource
getCollectionTag	GetCollectionTagParameter	List<TagResource>
deleteTag	DeleteTagParameter	void

##### Package

Informations about packages and contained classes in packages are listed unter [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.tag`

##### Package - Files

- TagResource.java
- EditTagParameter.java
- GetTagParameter.java

- GetCollectionTagParameter.java
- DeleteTagParameter.java

### 16.2.6 TagStoreTagList

#### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
getTagStoreTagList	GetTagStoreTagListParameter	TagStoreTagListResource
getCollectionTagStoreTagList	GetCollectionTagStoreTagListParam	List<TagStoreTagListResource>

#### Package

Informations about packages and contained classes in packages are listed unter [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.tagstoretaglist`

#### Package - Files

- TagStoreTagListResource.java
- GetTagStoreTagListParameter.java
- GetCollectionTagStoreTagListParameter.java

#### 16.2.6.1 TagStoreTag

#### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
getTagStoreTagListTagStoreTag	GetTagStoreTagParameter	TagStoreTagResource
getCollectionTagStoreTagListTagSto	GetCollectionTagStoreTagParameter	List<TagStoreTagResource>

#### Package

Informations about packages and contained classes in packages are listed unter [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.tagstoretaglist.tagstoretag`

#### Package - Files

- TagStoreTagResource.java
- GetTagStoreTagParameter.java
- GetCollectionTagStoreTagParameter.java

### 16.2.7 TagSuggestionList

#### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
createTagSuggestionList	CreateTagSuggestionListParameter	Long
getCollectionTagSuggestionList	GetCollectionTagSuggestionListPara	List<TagSuggestionListResource>

### Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.tagsuggestionlist`

### Package - Files

- TagSuggestionListResource.java
- CreateTagSuggestionListParameter.java
- GetCollectionTagSuggestionListParameter.java
- EF\_topicAccess.java

## 16.2.8 TimelineNote

### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
createTimelineNote	CreateTimelineNoteParameter	Long
editTimelineNote	EditTimelineNoteParameter	Long
getTimelineNote	GetTimelineNoteParameter	TimelineNoteResource
getCollectionTimelineNote	GetCollectionTimelineNoteParameter	List<TimelineNoteResource>

### Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.timelinenote`

### Package - Files

- TimelineNoteResource.java
- CreateTimelineNoteParameter.java
- EditTimelineNoteParameter.java
- GetTimelineNoteParameter.java
- GetCollectionTimelineNoteParameter.java
- EF\_topicAccess.java
- EUserNoteProperties.java
- ERights.java

#### 16.2.8.1 Property

### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
createProperty	CreatePropertyParameter	Long
getTimelineNoteProperty	GetPropertyParameter	PropertyResource
getCollectionTimelineNoteProperty	GetCollectionPropertyParameter	List<PropertyResource>

### Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.timelinenote.property`

### Package - Files

- PropertyResource.java
- CreatePropertyParameter.java
- GetPropertyParameter.java
- GetCollectionPropertyParameter.java

#### 16.2.8.2 TimelineNoteDiscussion

### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
getTimelineNoteTimelineNoteDiscussion	GetTimelineNoteDiscussionParameter	TimelineNoteDiscussionResource

### Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.timelinenote.timelinenotediscussion`

### Package - Files

- TimelineNoteDiscussionResource.java
- GetTimelineNoteDiscussionParameter.java

#### 16.2.8.3 TimelineNoteTag

### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
getTimelineNoteTimelineNoteTag	GetTimelineNoteTagParameter	TimelineNoteTagResource
getCollectionTimelineNoteTimelineNoteTag	GetCollectionTimelineNoteTagParameter	List<TimelineNoteTagResource>

## Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.timelinenote.timelinenotetag`

## Package - Files

- `TimelineNoteTagResource.java`
- `GetTimelineNoteTagParameter.java`
- `GetCollectionTimelineNoteTagParameter.java`

### 16.2.9 TimelineTag

#### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
<code>createTimelineTag</code>	<code>CreateTimelineTagParameter</code>	Long
<code>getCollectionTimelineTag</code>	<code>GetCollectionTimelineTagParameter</code>	List<TimelineTagResource>

## Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.timelinetag`

## Package - Files

- `TimelineTagResource.java`
- `CreateTimelineTagParameter.java`
- `GetCollectionTimelineTagParameter.java`
- `EF_topicAccess.java`

### 16.2.10 TimelineTopic

#### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
<code>createTimelineTopic</code>	<code>CreateTimelineTopicParameter</code>	Long
<code>getCollectionTimelineTopic</code>	<code>GetCollectionTimelineTopicParameter</code>	List<TimelineTopicResource>

## Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.timelinetopic`

## Package - Files

- `TimelineTopicResource.java`

- CreateTimelineTopicParameter.java
- GetCollectionTimelineTopicParameter.java
- EF\_topicAccess.java

### 16.2.11 TimelineUser

#### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
createTimelineUser	CreateTimelineUserParameter	Long
getCollectionTimelineUser	GetCollectionTimelineUserParameter	List<TimelineUserResource>

#### Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.timelineuser`

#### Package - Files

- TimelineUserResource.java
- CreateTimelineUserParameter.java
- GetCollectionTimelineUserParameter.java
- EF\_topicAccess.java

### 16.2.12 Topic

#### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
createTopic	CreateTopicParameter	Long
editTopic	EditTopicParameter	Long
getTopic	GetTopicParameter	TopicResource
getCollectionTopic	GetCollectionTopicParameter	List<TopicResource>
deleteTopic	DeleteTopicParameter	void

#### Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.topic`

#### Package - Files

- TopicResource.java
- CreateTopicParameter.java
- EditTopicParameter.java



- GetTopicParameter.java
- GetCollectionTopicParameter.java
- DeleteTopicParameter.java
- ETopicIdentifier.java
- ETopicListType.java

#### 16.2.12.1 ChildTopic

##### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
createChildTopic	CreateChildTopicParameter	Long
deleteChildTopic	DeleteChildTopicParameter	void

##### Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.topic.childtopic`

##### Package - Files

- ChildTopicResource.java
- CreateChildTopicParameter.java
- DeleteChildTopicParameter.java
- EParentTopicIdentifier.java

#### 16.2.12.2 ExternalObject

##### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
createExternalObject	CreateExternalObjectParameter	Long
editExternalObject	EditExternalObjectParameter	Long
getTopicExternalObject	GetExternalObjectParameter	ExternalObjectResource
getCollectionTopicExternalObject	GetCollectionExternalObjectParameter	List<ExternalObjectResource>
deleteExternalObject	DeleteExternalObjectParameter	void

##### Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.topic.externalobject`

##### Package - Files

- ExternalObjectResource.java

- CreateExternalObjectParameter.java
- EditExternalObjectParameter.java
- GetExternalObjectParameter.java
- GetCollectionExternalObjectParameter.java
- DeleteExternalObjectParameter.java
- ETopicIdentifier.java

### 16.2.12.3 Follow

#### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
createFollow	CreateFollowParameter	Long
getCollectionTopicFollow	GetCollectionFollowParameter	List<FollowResource>
deleteFollow	DeleteFollowParameter	void

#### Package

Informations about packages and contained classes in packages are listed unter [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.topic.follow`

#### Package - Files

- FollowResource.java
- CreateFollowParameter.java
- GetCollectionFollowParameter.java
- DeleteFollowParameter.java

### 16.2.12.4 Property

#### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
createProperty	CreatePropertyParameter	Long
getTopicProperty	GetPropertyParameter	PropertyResource
getCollectionTopicProperty	GetCollectionPropertyParameter	List<PropertyResource>

#### Package

Informations about packages and contained classes in packages are listed unter [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.topic.property`

#### Package - Files

- PropertyResource.java

- CreatePropertyParameter.java
- GetPropertyParameter.java
- GetCollectionPropertyParameter.java

#### 16.2.12.5 Right

##### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
editRight	EditRightParameter	Long
getTopicRight	GetRightParameter	RightResource

##### Package

Informations about packages and contained classes in packages are listed unter [Important Classes](#).

com.communote.plugins.api.rest.client.resource.topic.right

##### Package - Files

- RightResource.java
- EditRightParameter.java
- GetRightParameter.java

#### 16.2.12.6 Role

##### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
createRole	CreateRoleParameter	Long
editRole	EditRoleParameter	Long
getTopicRole	GetRoleParameter	RoleResource
getCollectionTopicRole	GetCollectionRoleParameter	List<RoleResource>
deleteRole	DeleteRoleParameter	void

##### Package

Informations about packages and contained classes in packages are listed unter [Important Classes](#).

com.communote.plugins.api.rest.client.resource.topic.role

##### Package - Files

- RoleResource.java
- CreateRoleParameter.java
- EditRoleParameter.java
- GetRoleParameter.java

- GetCollectionRoleParameter.java
- DeleteRoleParameter.java
- ETopicIdentifier.java
- ERole.java

#### 16.2.12.7 RoleBulkExternal

##### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
createRoleBulkExternal	CreateRoleBulkExternalParameter	Long
editRoleBulkExternal	EditRoleBulkExternalParameter	Long
getTopicRoleBulkExternal	GetRoleBulkExternalParameter	RoleBulkExternalResource
deleteRoleBulkExternal	DeleteRoleBulkExternalParameter	void

##### Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.topic.rolebulkexternal`

##### Package - Files

- RoleBulkExternalResource.java
- CreateRoleBulkExternalParameter.java
- EditRoleBulkExternalParameter.java
- GetRoleBulkExternalParameter.java
- DeleteRoleBulkExternalParameter.java
- ETopicIdentifier.java

#### RoleBulkExternalTopicRight

##### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
getTopicRoleBulkExternalRoleBulkE	GetRoleBulkExternalTopicRightPar	RoleBulkExternalTopicRightResource

##### Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.topic.rolebulkexternal.rolebulkexternalt`

##### Package - Files

- RoleBulkExternalTopicRightResource.java
- GetRoleBulkExternalTopicRightParameter.java
- ERole.java

## 16.2.12.8 RoleExternal

**Methods**

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
createRoleExternal	CreateRoleExternalParameter	Long
editRoleExternal	EditRoleExternalParameter	Long
getTopicRoleExternal	GetRoleExternalParameter	RoleExternalResource
getCollectionTopicRoleExternal	GetCollectionRoleExternalParameter	List<RoleExternalResource>
deleteRoleExternal	DeleteRoleExternalParameter	void

**Package**

Informations about packages and contained classes in packages are listed unter [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.topic.roleexternal`

**Package - Files**

- RoleExternalResource.java
- CreateRoleExternalParameter.java
- EditRoleExternalParameter.java
- GetRoleExternalParameter.java
- GetCollectionRoleExternalParameter.java
- DeleteRoleExternalParameter.java
- ETopicIdentifier.java
- ERole.java

## 16.2.13 User

**Methods**

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
createUser	CreateUserParameter	Long
editUser	EditUserParameter	Long
getUser	GetUserParameter	UserResource
getCollectionUser	GetCollectionUserParameter	List<UserResource>

**Package**

Informations about packages and contained classes in packages are listed unter [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.user`

**Package - Files**

- UserResource.java

- CreateUserParameter.java
- EditUserParameter.java
- GetUserParameter.java
- GetCollectionUserParameter.java

#### 16.2.13.1 ExternalLogin

##### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
getCollectionUserExternalLogin	GetCollectionExternalLoginParamete	List<ExternalLoginResource>

##### Package

Informations about packages and contained classes in packages are listed unter [Important Classes](#).

com.communote.plugins.api.rest.client.resource.user.externallogin

##### Package - Files

- ExternalLoginResource.java
- GetCollectionExternalLoginParameter.java

##### ExternalProperty

##### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
getUserExternalLoginExternalProper	GetExternalPropertyParameter	ExternalPropertyResource

##### Package

Informations about packages and contained classes in packages are listed unter [Important Classes](#).

com.communote.plugins.api.rest.client.resource.user.externallogin.externalproperty

##### Package - Files

- ExternalPropertyResource.java
- GetExternalPropertyParameter.java

#### 16.2.13.2 Follow

##### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
createFollow	CreateFollowParameter	Long

getCollectionUserFollow	GetCollectionFollowParameter	List<FollowResource>
deleteFollow	DeleteFollowParameter	void

## Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.user.follow`

### Package - Files

- FollowResource.java
- CreateFollowParameter.java
- GetCollectionFollowParameter.java
- DeleteFollowParameter.java

#### 16.2.13.3 Image

### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
getUserImage	GetImageParameter	ImageResource

## Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.user.image`

### Package - Files

- ImageResource.java
- GetImageParameter.java
- ESize.java

#### 16.2.13.4 NavigationItem

### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
createNavigationItem	CreateNavigationItemParameter	Long
editNavigationItem	EditNavigationItemParameter	Long
getUserNavigationItem	GetNavigationItemParameter	NavigationItemResource
getCollectionUserNavigationItem	GetCollectionNavigationItemParameter	List<NavigationItemResource>
deleteNavigationItem	DeleteNavigationItemParameter	void

## Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.user.navigationitem`

#### Package - Files

- `NavigationItemResource.java`
- `CreateNavigationItemParameter.java`
- `EditNavigationItemParameter.java`
- `GetNavigationItemParameter.java`
- `GetCollectionNavigationItemParameter.java`
- `DeleteNavigationItemParameter.java`

#### 16.2.13.5 Observation

##### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
<code>getUserObservation</code>	<code>GetObservationParameter</code>	<code>ObservationResource</code>
<code>getCollectionUserObservation</code>	<code>GetCollectionObservationParameter</code>	<code>List&lt;ObservationResource&gt;</code>

#### Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.user.observation`

#### Package - Files

- `ObservationResource.java`
- `GetObservationParameter.java`
- `GetCollectionObservationParameter.java`

#### 16.2.13.6 Property

##### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
<code>createProperty</code>	<code>CreatePropertyParameter</code>	<code>Long</code>
<code>getUserProperty</code>	<code>GetPropertyParameter</code>	<code>PropertyResource</code>
<code>getCollectionUserProperty</code>	<code>GetCollectionPropertyParameter</code>	<code>List&lt;PropertyResource&gt;</code>

#### Package

Informations about packages and contained classes in packages are listed under [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.user.property`

#### Package - Files



- PropertyResource.java
- CreatePropertyParameter.java
- GetPropertyParameter.java
- GetCollectionPropertyParameter.java

#### 16.2.13.7 UserPreference

##### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
editUserPreference	EditUserPreferenceParameter	Long
getUserUserPreference	GetUserPreferenceParameter	UserPreferenceResource

##### Package

Informations about packages and contained classes in packages are listed unter [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.user.userpreference`

##### Package - Files

- UserPreferenceResource.java
- EditUserPreferenceParameter.java
- GetUserPreferenceParameter.java

##### Preference

##### Methods

Please read the appropriate documentation part of [Resources](#), for an specific information about the attributs an parameters in every parameter class.

Name	Parameter	Return
createPreference	CreatePreferenceParameter	Long
editPreference	EditPreferenceParameter	Long
getUserUserPreferencePreference	GetPreferenceParameter	PreferenceResource
deletePreference	DeletePreferenceParameter	void

##### Package

Informations about packages and contained classes in packages are listed unter [Important Classes](#).

`com.communote.plugins.api.rest.client.resource.user.userpreference.preference`

##### Package - Files

- PreferenceResource.java
- CreatePreferenceParameter.java
- EditPreferenceParameter.java
- GetPreferenceParameter.java
- DeletePreferenceParameter.java

## 17 Imprint

---

**Publisher:**

Communote GmbH  
Kleiststraße 10a  
D-01129 Dresden/Germany

Phone: +49.351.83382.100  
Fax: +49.351.83382.199

[info@communote.com](mailto:info@communote.com)  
[www.communote.com](http://www.communote.com)