

Inter-Applikation Interface (http-api)

Inter-Applikation Interface (httpapi)

This chapter describes the API function as it is used to access the subscriber's database. BACKCLICK provides it as an alternate way to execute subscriptions, unsubscriptions, or changes on the subscriber's profile, by using the "http" records. The API is used for communication between two server procedures. It does not provide readable output, only status codes.

If a subscription page (e.g. for customer data, a shop system) already exists, it is possible to generate a link (URL) filled with this data, and add the customers to your BACKCLICK database. The data will then be directly broadcast from the application to your BACKCLICK system via the "http." The advantage here is that you only need one subscription page (application) but can use this data in more than one system. And your applications can communicate with BACKCLICK directly on the record level.

Because of the mandator management of BACKCLICK, it is necessary to hand over the mandator's ID with every call of the http api. If a mandatory ID is handed over, BACKCLICK will assign the call to the mandator with the ID=0.

! PLEASE NOTE !

This interface may be called without any restrictions by servers as well as clients. To restrict these calls, you may enter IP addresses that are exclusively allowed to start a call (System Administration -> Basic Settings -> WebInterface). Therefore, you need to mark the check box in front of the "HTTP subscriber API" to activate the text area on the righthand side. You may then enter the IP addresses.

The following functions are supported by this API:

- Subscription of new subscribers on to every mailing list/newsletter
- Changes in the subscriber's profile, including the change of the e-mail address
- Additional order of newsletters
- Unsubscription of existing subscribers (completely or only for some of the mailing lists)

Basics

Every function is called by the http record. BACKCLICK will return a status code on every call. The URL for the communication is built up as described

here:

`http://domain.customer.de/bc/servlet/web.httpapi?..`

The following values are accepted parameters:

- EMAIL
- NEWEMAIL
- NEWSLETTER (ID of the mailing list(s))
- PASSWORD
- CHANGETYPE (only for changes in profile, additive change = 1, absolute change = 0)
- SENDCONFIRM (sends out confirmation of un-/subscription = 1, sends out no confirmation of un-/subscription = 0)
- XXXXXXX (all of the existing variables (Names), defined in the database designer)
- DELETE (no NEWSLETTER chosen, subscriber will be removed = 1, selected NEWSLETTER/mailing list IDs, subscriber will be removed from the selected IDs=0)
- MAKENUPDATE (prevents the overwriting/extension of subscriber's data if the subscriber exists = 1, Default = 0)
- DOI-TEMPLATE (ID of the double opt in webinterface template, that shall be used for new subscriptions. If you pass a doi template id here, the subscriber gets a confirmation email for new subscriptions)

Default parameters are always EMAIL and NEWSLETTER (only if a subscriber should be removed; NEWSLETTER will then be changed into DELETE=1). All other parameters defined in the database designer are optional parameters.

Subscription of new subscribers on every mailing list

Example:

```
http://domain.customer.de/bc/servlet/web.httpapi?EMAIL=abc@def.gh&LASTNAME=Jones&FIRSTNAME=Robert&NEWSLETTER=1;2&MID=0
```

This example shows that it is possible to enter information to into the data fields (here, FIRSTNAME, LASTNAME).

Assignments on mailing lists (NEWSLETTER) need to be separated by a semicolon if there is more than one assignment.

If you want to send a confirmation mail to the subscriber for a double opt in, you have to pass the URL parameter DOI-TEMPLATE. The value of this parameter is the ID of a double opt in webinterface template, configured in your backclick system. (You can find more information about double opt in webinterface templates here: [Subscription \(Double-opt-in\)](#))

Change of subscriber's profile, including change of e-mail address

The URL should be the same as if you

were creating a subscriber. If BACKCLICK detects that there already is a subscriber owning the data included in the URL, BACKCLICK will change the profile and add the newly won data. In the example below you will see that the subscriber additionally got a password, and the assignment on the mailing list has been changed.

There are two special cases in reference to changing the subscriber's data:

1. The changing of an e-mail address needs a special parameter: NEWEMAIL
2. The assignment on to mailing lists may be done absolutely or additively. To assign mailing lists additively, you need to replenish the list of parameters with the parameter CHANGETYPE=1.

Changing the e-mail address:

```
http://domain.customer.de/bc/servlet/web.httpapi?EMAIL=abc@def.gh&LASTNAME=Jones&FIRSTNAME=Robert&NEWEMAIL=xyz@xyz.zz&MID=0
```

Changing the data to an absolute assignment on mailing lists:

```
http://domain.customer.de/bc/servlet/web.httpapi?EMAIL=abc@def.gh&LASTNAME=Jones&FIRSTNAME=Robert&PASSWORD=123456&NEWSLETTER=1;2;3&MID=0
```

The subscriber is assigned to the mailing lists with the IDs 1, 2, and 3. All of the other assignments will be erased.

Changing the data to an additive assignment on mailing lists:

```
http://domain.customer.de/bc/servlet/web.httpapi?EMAIL=abc@def.gh&LASTNAME=Jones&FIRSTNAME=Robert&PASSWORD=123456&NEWSLETTER=4;6&CHANGETYPE=1&MID=0
```

The subscriber is assigned to the mailing lists with the IDs 4 and 6. This is an additional assignment that does not erase all of the other assignments to other mailing lists.

Remove a subscriber

You can remove all of the subscriber's data from your system, as well as only their assignments to one or more mailing lists. In every case, you must name the e-mail address of the subscriber who should be removed, as well as the parameter DELETE=1.

```
http://domain.customer.de/bc/servlet/web.httpapi?EMAIL=abc@def.gh&DELETE=1&MID=0
```

In this case, the subscriber's information, including all his or her data and the assignments, will be removed from the system.

```
http://domain.customer.de/bc/servlet/web.httpapi?EMAIL=abc@def.gh&NEWSLETTER=1&DELETE=1&MID=0
```

In this case, ONLY the assignment on mailing list ID 1 will be removed.

Zusätzliche Parameter

It is also possible to add the parameter SENDCONFIRM, which will cause BACKCLICK to send "Welcome" and "Goodbye" messages. You will configure the text of these e-mails while creating the corresponding mailing lists. They will be sent in response to subscriptions and unsubscriptions.

Example:

```
http://domain.customer.de/bc/servlet/web.httpapi?EMAIL=abc@def.gh&DELETE=1&SENDCONFIRM=1&MID=0
```

In this case, BACKCLICK will send a confirmation e-mail to every subscriber who unsubscribes from one or more mailing lists. Of course, this assumes that this message is defined for the mailing list(s).

Messages on status of the API

Error codes:

- 1 --> entered e-mail address is syntactically wrong (RFC e-mail check failed)
- 2 --> new e-mail address incorrect while changing the subscriber's e-mail (RFC e-mail check failed)
- 3 --> no e-mail address entered
- 4 --> no parameter NEWSLETTER existing in URL
- 5 --> e-mail address does not exist in system while changing subscriber's e-mail address
- 6 --> no connection to the database between API server and DBMS at present
- 7 --> handed-over parameter NEWSLETTER is either empty or entered list ID does not exist
- 8 --> no password entered, wrong password entered
- 9 --> subscriber is listed on the internal eRobinson list of BACKCLICK
- 10 --> function deactivated because of license reasons
- 11 --> error while unsubscribing, may try again
- 12 --> the unsubscribing e-mail address is not subscribed to any mailing list
- 100 --> HTTP-API function deactivated in the system settings; please activate

Success codes:

- OK2 --> Subscriber removed successfully
- OK3 --> Subscriber removed successfully
- OK4 --> Subscriber saved successfully
- OK5 --> Subscriber's data changed successfully