

CONSUMER DATA PLATFORM PLAYBOOK

POSTBACK API CONFIGURATION V1.4

MARTECH - CONSUMER JOURNEY

AB-INBEV GLOBAL TEAM



CDP

Consumer Data Platform

How to implement and configure the CDP postback-api in a digital product

Goal

The goal of this documentation is to explain how to implement the postback-api in a digital product to start collecting the consumer data coming from web forms, with consents for each purpose, following the GDPR requirements.

Collecting consumer data from webform submission

Form submission is not collected by default since different websites implement the submit event differently. That's why you'll need to read all this document to understand the best method to connect your application to CDP.

How to connect a website with CDP

In order to collect the consumer declared data (PII) from your webforms, you will need implement a Rest-API connection between your website and CDP.

Assumptions

We assume that the user is trying to bring event data (1) one by one and (2) asynchronously.

- 1. One by one:** meaning that the user event is created based on user action, such as web form submits.
- 2. Asynchronously:** meaning that this API needs to return immediately in order for the calling application to proceed with other tasks. This means that the 200 response means that the event data was successfully received, **NOT** successfully processed.



CDP

Consumer Data Platform

Technical Details

The full API specification can be found on the official Treasure Data documentation [here](#).

- It supports both GET and POST;
- For both GET and POST, it returns HTTP status 200 if the payload is well-formatted;
- In particular, it does NOT check if API key is correct. So please double check that you have the correct API Key.

Specifically for ABI, you need to set the right `DATABASE` and `TABLE` values in <https://in.treasuredata.com/postback/v3/event/DATABASE/TABLE>.

Generating the Postback API using the CDP Codes Generator tool

We created a tool where you can generate the postback API codes automatically. Please click [here](#) to access.

To generate a new postback api code, you must have been created previously the CDP JS tag in the same tool. If you did not create the tag yet, please follow the steps described on this [playbook here](#).

If you already have the tag created just follow the step by step below:

- **category:** choose the category that describes best your website. The description of each category you can find below:
 - Beer Brand Website: for beer brand pages
 - NAB Brand Website: for non alcoholic brands (Ex: Pepsi)
 - Other Brand Website: for near beer brands (Ex: Skol Beats / Spiked Seltzer);
 - Institutional Website: for the institutional ones like ab-inbev.com
 - Experiential Website: specific for experiential campaigns
 - Promotional Website: for promotional campaigns
 - Ecommerce
 - Loyalty Program
- **form name:** what is the name of the form. Example: register, contact us, newsletter, subscription etc.
- **campaignName:** if the website belongs to a specific campaign or is a hot-site, you can inform the campaign name (i.e.: Superbowl 2020, Skol Verão, Bud PL).



CDP

Consumer Data Platform

This field is optional.;

- **campaignId:** if you campaign has a specific ID, you can inform in this field. Its also optional
- **technology:** select the technology of your back end. The generator works for PHP and NodeJS.
- **form fields:** select on the list what fields you have in your webform. If you can't find your field on the list, just click on "Contact Us" and send a email to consumerdatasquad@ab-inbev.com
- **unify:** this box is active by default. You **only** need to deselect this if your digital product collects consumer data in a single device only. Ex: promotional campaigns with promoters collecting consumer data in tablets. If your website is on the internet and anyone can access, leave this box marked.
- **Terms and conditions:** you must need to select if the Terms and Conditions acceptance are informed or not informed by your consumer.
- **Marketing Activation:** you must need to select if the Marketing Activation consent acceptance is informed or not informed.

After fill all fields the system will automatically generate the Postback API code for you like the print below:

Backend integration code

CDP Postback API

Mandatory Fields

Just a recap of the mandatory fields that needs to be on your postback API integration code. If you generated the code using the CDP Code Generator all of them will be covered.

In the body of the request, you must meet the following requirements:

1. **abi_brand** for the brand name of the URL;
2. **abi_form** for the webform name. For example, in the later example PHP, this value is set as **newsletter**, that is the div ID informed in the HTML;
3. **abi_campaign** for the webform campaign name. This is **OPTIONAL**, so if your website is not related to specific campaign, you can leave this field empty;
4. **abi_email** for the user's email **OR** **abi_phone** for the user's phone number;



CDP

Consumer Data Platform

5. `td_client_id` for the user's 1st party cookie ID. This is the value of `td` cookie on the website. You need to parse this value from the cookies sent to your backend. For example, in the later example PHP code, this value is read from `$_COOKIE` variable;
6. `td_url` for the URL of the page. For example, in the later example PHP code, this value is read from `$_SERVER` variable's `HTTP_HOST` + `REQUEST_URI` fields;
7. `td_host` for the domain of the page. For example, in the later example PHP, this value is read from `$_SERVER` variable's `HTTP_HOST` field;
8. `purpose_name` for the consent field Marketing Activation (**MARKETING-ACTIVATION**), Terms & Conditions and Privacy Policy (**TC-PP**) and Marketing Analytics (**MARKETING-ANALYTICS**) opt-in. This should be captured as an array of all purposes that the user has opted-in for. If they are not informed than the array should bring `TC-PP-NOTINFORMED` and `MARKETING-ACTIVATION-NOTINFORMED`. Never leave this array empty.
9. `td_import_method = postback-api-1.2` the value of this field should be always `postback-api`

*****Note:** If you have any other kind of consent in you webform, you can send using the same logic as the **mandatory** ones, with **previously alignment** with the Consumer Data Squad Team at consumerdatasquad@ab-inbev.com.

Example cURL Requests

Zone Name	cURL example
South America (SAZ)	<pre>curl -D a -X POST -H 'Content-Type: application/json' -H 'X-TD-Write-Key: 10086/9c06ed6fa48e0fb6952ed42773cca1cc1d43684e' --data-binary '{"abi_brand": "brahma", "abi_form": "cadastro", "abi_campaign": "ClubeN1", "abi_email": "kiyoto@treasuredata.com", "td_client_id": "6900c3fb-51dd-498f-b003-0ecdbf57d05d", "td_url": "https://www.brahma.com.br/clube-n1/cadastro", "td_host": "www.brahma.com.br", "purpose_name": ['MARKETING-ACTIVATION', 'TC-PP']}' https://in.treasuredata.com/postback/v3/event/saz_source/bra_web_form</pre>
North America (NAZ)	<pre>curl -D a -X POST -H 'Content-Type: application/json' -H 'X-TD-Write-Key: 10086/9c06ed6fa48e0fb6952ed42773cca1cc1d43684e' --data-binary '{"abi_brand": "brahma", "abi_form": "cadastro", "abi_campaign": "ClubeN1", "abi_email": "kiyoto@treasuredata.com", "td_client_id": "6900c3fb-51dd-498f-b003-0ecdbf57d05d", "td_url": "https://www.brahma.com.br/clube-n1/cadastro", "td_host": "www.brahma.com.br", "purpose_name": ['MARKETING-ACTIVATION', 'TC-PP']}' https://in.treasuredata.com/postback/v3/event/naz_source/usa_web_form</pre>

**CDP**

Consumer Data Platform

Zone Name	cURL example
Middle America (MIDAM)	<pre>curl -D a -X POST -H 'Content-Type: application/json' -H 'X-TD-Write-Key: 10086/9c06ed6fa48e0fb6952ed42773cca1cc1d43684e' --data-binary '{"abi_brand": "brahma", "abi_form": "cadastro", "abi_campaign": "ClubeN1", "abi_email": "kiyoto@treasuredata.com", "td_client_id": "6900c3fb-51dd-498f-b003-0ecdbf57d05d", "td_url": "https://www.brahma.com.br/clube-n1/cadastro", "td_host": "www.brahma.com.br", "purpose_name": ['MARKETING-ACTIVATION', 'TC-PP']}'</pre> https://in.treasuredata.com/postback/v3/event/midam_source/col_web_form
Africa	<pre>curl -D a -X POST -H 'Content-Type: application/json' -H 'X-TD-Write-Key: 10086/9c06ed6fa48e0fb6952ed42773cca1cc1d43684e' --data-binary '{"abi_brand": "brahma", "abi_form": "cadastro", "abi_campaign": "ClubeN1", "abi_email": "kiyoto@treasuredata.com", "td_client_id": "6900c3fb-51dd-498f-b003-0ecdbf57d05d", "td_url": "https://www.brahma.com.br/clube-n1/cadastro", "td_host": "www.brahma.com.br", "purpose_name": ['MARKETING-ACTIVATION', 'TC-PP']}'</pre> https://in.treasuredata.com/postback/v3/event/africa_source/zaf_web_form
Europe	<pre>curl -D a -X POST -H 'Content-Type: application/json' -H 'X-TD-Write-Key: 10086/9c06ed6fa48e0fb6952ed42773cca1cc1d43684e' --data-binary '{"abi_brand": "brahma", "abi_form": "cadastro", "abi_campaign": "ClubeN1", "abi_email": "kiyoto@treasuredata.com", "td_client_id": "6900c3fb-51dd-498f-b003-0ecdbf57d05d", "td_url": "https://www.brahma.com.br/clube-n1/cadastro", "td_host": "www.brahma.com.br", "purpose_name": ['MARKETING-ACTIVATION', 'TC-PP']}'</pre> https://in.treasuredata.com/postback/v3/event/eur_source/gbr_web_form
Asia Pacific	<pre>curl -D a -X POST -H 'Content-Type: application/json' -H 'X-TD-Write-Key: 10086/9c06ed6fa48e0fb6952ed42773cca1cc1d43684e' --data-binary '{"abi_brand": "brahma", "abi_form": "cadastro", "abi_campaign": "ClubeN1", "abi_email": "kiyoto@treasuredata.com", "td_client_id": "6900c3fb-51dd-498f-b003-0ecdbf57d05d", "td_url": "https://www.brahma.com.br/clube-n1/cadastro", "td_host": "www.brahma.com.br", "purpose_name": ['MARKETING-ACTIVATION', 'TC-PP']}'</pre> https://in.treasuredata.com/postback/v3/event/apac_source/aus_web_form

**CDP**

Consumer Data Platform

APPENDIX

Quick start with PHP

Here's an example code for PHP:

```
<?php

/*
This example has been tested in the following environment

$ php --version
PHP 7.1.23 (cli) (built: Feb 22 2019 22:19:32) ( NTS )
Copyright (c) 1997-2018 The PHP Group
Zend Engine v3.1.0, Copyright (c) 1998-2018 Zend Technologies

*/

private function __sendTD($form_data, $country, $brand, $campaign, $form, $production){

    $td_env = $production ? 'prod' : 'dev';

    // This is your HTML form data and any other data you want to send to TD

    $http_protocol = isset($_SERVER['https']) ? 'https://' : 'http://';

    $form_data['abi_brand'] = $brand;
    $form_data['abi_campaign'] = $campaign;
    $form_data['abi_form'] = $form;
    $form_data['td_import_method'] = 'postback-api';
    $form_data['td_client_id'] = $_COOKIE['_td'];
    $form_data['td_url'] = $http_protocol . $_SERVER['HTTP_HOST'] . $_SERVER['REQUEST_URI'];
    $form_data['td_host'] = $_SERVER['HTTP_HOST'];

    $td_country = $country;

    $td_apikey = $td_env !== 'prod' ? '9648/41e45454b77308046627548e0b4fe2ddbc0893d2' : '10086/9c06ed6fa48e0fb6952ed42773cca1cc1d43684e'; // Your TD API key. DO NOT CHANGE

    // The mapping from country name to zone name. DO NOT MODIFY
```

**CDP**

Consumer Data Platform

```
$country_zone_mapping = array("nga"=>"africa", "zwe"=>"africa", "zaf"=>"africa", "aus"=>"apac", "chn"=>"apac", "ind"
=>"apac", "jpn"=>"apac", "kor"=>"apac", "tha"=>"apac", "vnm"=>"apac", "bel"=>"eur", "fra"=>"eur", "deu"=>"eur", "
ita"=>"eur", "nld"=>"eur", "rus"=>"eur", "esp"=>"eur", "ukr"=>"eur", "gbr"=>"eur", "col"=>"midam", "dom"=>"midam"
, "ecu"=>"midam", "slv"=>"midam", "gtm"=>"midam", "hnd"=>"midam", "mex"=>"midam", "pan"=>"midam", "per"=>"midam",
"can"=>"naz", "usa"=>"naz", "arg"=>"saz", "bol"=>"saz", "bra"=>"saz", "chl"=>"saz", "ury"=>"saz");

$td_zone = $country_zone_mapping[$td_country];
$curl = curl_init();

$curl_opts = array(
    CURLOPT_URL => "https://in.treasuredata.com/postback/v3/event/{$td_zone}_source/{$td_country}_web_form",
    CURLOPT_POST => true,
    CURLOPT_RETURNTRANSFER => true,
    CURLOPT_HTTPHEADER => array(
        'Content-Type: application/json',
        "X-TD-Write-Key: ${td_apikey}"
    ),
    CURLOPT_POSTFIELDS => json_encode($form_data)
);

curl_setopt_array($curl, $curl_opts);

$response = @curl_exec($curl);
$response_code = curl_getinfo($curl, CURLINFO_HTTP_CODE);

curl_close($curl);

return $response_code;
}

// define variable that will be used to tell the __sendTD method if it should send to the production database
$is_production = preg_match("@brahma.com.br@", $_SERVER['SERVER_NAME']);

// define the purpose variable as an empty array
$purposes = array();

// check whether the TC-PP checkbox is checked, and if it is, then adds it to the purpose array
if($tcpp)
    $purposes[] = 'TC-PP';

// check whether the MARKETING-ACTIVATION checkbox is checked, and if it is, then adds it to the purpose array
if($optin)
    $purposes[] = 'MARKETING-ACTIVATION';

// here it's possible to add additional purposes to the purpose array
```


**CDP**

Consumer Data Platform

```
// runs the __sendTD method with parameters got from the request, it should be changed based on your form fields,
// country, brand, campaign, form, and whether if it's running in the production environment or not
// in this particular case we're sending data collected from the Newsletter form, which is not under a campaign, country
// bra (Brazil), brand Brahma, and the form had 6 input fields (name, email, state, city, gender, data of birth) and
// 2 checkboxes (tc-pp and marketing activation)
$tdstatus = $this->__sendTD(
    array(
        'abi_name' => $name,
        'abi_email' => $email,
        'abi_state' => $state,
        'abi_city' => $city,
        'abi_gender' => $gender,
        'abi_dateofbirth' => date('Y-m-d', $real_date_time),
        'purpose_name' => $purposes,
    ),
    // form data
    'bra',
    // country - Always use three letters for country name
    'Brahma',
    // brand
    null,
    // campaign
    'Newsletter',
    // form
    $is_production
    // production flag
);
```

Quick start with NODEJS

Here's an example code for NODEJS:

```
const __sendTD = function(form_data, country, brand, campaign, form, production, client_id, url, host){

    return new Promise(function(resolve, reject){

        const td_env = production ? 'prod' : 'dev'

        form_data['td_import_method'] = 'postback-api'
        form_data['td_client_id'] = client_id
        form_data['td_url'] = url
        form_data['td_host'] = host

        const td_country = country

        const td_apikey = $td_env !== 'prod' ? '9648/41e45454b77308046627548e0b4fe2ddbc0893d2' : '10086/9c06ed6fa48e0fb6952ed42773cca1cc1d43684e';

        const country_zone_mapping = {"nga":"africa", "zwe":"africa", "zaf":"africa", "aus":"apac", "chn":"apac", "ind":"apac", "jpn":"apac", "kor":"apac", "tha":"apac", "vnm":"apac", "bel":"eur", "fra":"eur", "deu":"eur", "ita":"eur", "nld":"eur", "rus":"eur", "esp":"eur", "ukr":"eur", "gbr":"eur", "col":"midam", "dom":"midam", "ecu":"midam",
```

**CDP**

Consumer Data Platform

```
"slv":"midam", "gtm":"midam", "hnd":"midam", "mex":"midam", "pan":"midam", "per":"midam", "can":"naz", "usa":"naz", "arg":"saz", "bol":"saz", "bra":"saz", "chl":"saz", "ury":"saz"}
```

```
const td_zone = country_zone_mapping[td_country]
```

```
fetch(`https://in.treasuredata.com/postback/v3/event/${td_zone}_source/${td_country}_web_form`, {
  method: 'POST',
  headers: {
    'Accept': 'application/json',
    'Content-Type': 'application/json',
    'X-TD-Write-Key': td_apikey
  },
  body: JSON.stringify(form_data)
}).then(r => {
```

```
  if(r.ok)
    r.json().then(data => {
      resolve(data)
    })
  else
    reject(new Error('Response from server was not ok'))
```

```
}).catch(e => {
  reject(e)
})
```

```
})
```

```
}
```

```
// Example of implementation
```

```
// This implementation assumes that it's using ExpressJS with Cookie-Parser
```

```
import moment from 'moment'
```

```
// check if is on production server
```

```
const is_production = false // change this according to its environment
```

```
const host = req.get('host')
```

```
const url = req.protocol + '://' + req.get('host') + req.originalUrl
```

```
const client_id = req.cookies['_td']
```

```
// define the purpose variable as an empty array
```

```
let purposes = []
```

**CDP**

Consumer Data Platform

```
// check whether the TC-PP checkbox is checked, and if it is, then adds it to the purpose array
if(tcpp)
  purposes.push('TC-PP')

// check whether the MARKETING-ACTIVATION checkbox is checked, and if it is, then adds it to the purpose array
if(optin)
  purposes.push('MARKETING-ACTIVATION')

// here it's possible to add additional purposes to the purpose array
// runs the __sendTD method with parameters got from the request, it should be changed based on your form fields, country,
// brand, campaign, form, and whether it's running in the production environment or not
// in this particular case we're sending data collected from the Newsletter form, which is not under a campaign,
// country bra (Brazil), brand Brahma, and the form had 6 input fields (name, email, state, city, gender, data of birth)
// and 2 checkboxes (tc-pp and marketing activation)

const tdstatus = __sendTD(
  {
    'abi_name': name,
    'abi_email': email,
    'abi_state': state,
    'abi_city': city,
    'abi_gender': gender,
    'abi_dateofbirth': moment().format('YYYY-MM-DD'),
    'purpose_name': purposes,
  },
  // form data
  'bra',
  // country - Always use three letters for country name
  'Brahma',
  // brand
  null,
  // campaign
  'Newsletter',
  // form
  is_production,
  // production flag
  client_id,
  // _td cookie
  url,
  // current url
  host,
  // host
)
```

Version	Author	Date	Description of Version
			Main or important changes since previous version
V1.1	Panayote Neto	December 17 th , 2019	Final version of CDP Implementation playbook v.5
V1.2	Panayote Neto	January 20 th , 2020	Added NODEJS code example in Appendix section



CDP

Consumer Data Platform

V1.3	Luiz Gama	Mach 21, 2020	Changes on text
V1.4	Luiz Gama	May 28, 2020	Including Code Generator Tool instructions