

Baskettracking (Serversided)

- [Einführung](#)
 - [Überblick](#)
- [Technische Grundlagen](#)
 - [Funktionsweise](#)
 - [Endpoint & Parameter](#)
- [JSON-Struktur](#)
 - [JSON-Struktur](#)
- [Implementierung](#)
 - [PHP Implementierung](#)
- [Voucher](#)
 - [Neue Seite](#)
- [Umsetzung im Falle von campaign_alias](#)

Einführung

Überblick

Das **Baskettracking** ermöglicht es, Transaktionen mitsamt Warenkorb-Details (**Basket**) über einen **serverseitigen Trackingaufruf** in das System einzuspielen.

Im Gegensatz zu clientseitigen Methoden (z. B. Pixeltracking) erfolgt hier die Übertragung der Daten direkt vom **Server des Advertisers** an das System von **easy.MARKETING**.

Warum Baskettracking?

- **Präzisere Attribution:** Jeder Artikel im Warenkorb kann einzeln übergeben werden.
 - **Serverseitige Sicherheit:** Tracking findet unabhängig vom Browser oder möglichen Adblockern statt.
 - **Flexibilität:** Neben den Pflichtfeldern können zusätzliche Parameter (z. B. Gutscheincodes) übergeben werden.
-

Die Rolle der EMID

Für die **Attribution** ist die sogenannte **EMID** (easy.MARKETING-ID) entscheidend.

- Sie wird beim Klick über den Click-Link an die Landingpage übergeben.
 - Der Advertiser-Server muss diese ID beim serverside-Call wieder an das System zurückspielen.
 - Ohne EMID ist keine vollständige Customer Journey nachvollziehbar.
-

Technische Grundlagen

Funktionsweise

Das Baskettracking basiert auf einem **serverseitigen Aufruf** vom Shop-System (Advertiser) an das Tracking-System von **easy.MARKETING**.

Dabei werden **Transaktions- und Artikeldaten** in einem JSON-Array übergeben. Jedes bestellte Produkt entspricht einem **Basket-Item**, das mit den relevanten Informationen (z. B. Preis, Anzahl, Produktname) befüllt wird.

Ablauf im Überblick

1. Klick auf Werbemittel

- Ein Nutzer klickt auf ein Werbemittel (z. B. Banner oder Link).
- Das System hängt die **EMID** (Tracking-ID) an die Ziel-URL an.

2. Besuch der Landingpage

- Die EMID wird vom Advertiser-System (Frontend oder Backend) gespeichert.
- Dadurch ist die Zuordnung des Nutzers sichergestellt.

3. Bestellung im Shop

- Der Nutzer führt eine Transaktion durch.
- Das Shop-System erstellt eine Liste aller bestellten Produkte (Basket).

4. Serverside Tracking-Call

- Das Shop-System baut einen Trackingaufruf auf und übergibt:
 - Kampagnen- und Triggerdaten
 - Bestellnummer (Token)
 - Basket-Daten (Artikel, Preis, Menge, Kategorie, etc.)
 - Die gespeicherte **EMID**
 - Der Aufruf erfolgt an den **Tracking-Endpoint** von easy.MARKETING.
-

Rolle der EMID

Die **EMID** ist eine eindeutige Kennung, die für die **Attribution der Transaktion** notwendig ist:

- Sie wird **beim Klick** erzeugt und an den Advertiser übergeben.
- Sie muss **beim Conversion-Aufruf** mitgesendet werden.
- Ohne EMID kann die Customer Journey **nicht korrekt zugeordnet** werden.

Endpoint & Parameter

Damit Baskettracking funktioniert, müssen die Daten vom Shop-System an den entsprechenden **Tracking-Endpoint** von easy.MARKETING übermittelt werden. Die Übergabe erfolgt per **HTTP-GET-Request** mit einem JSON-Array im Parameter `json`.

Endpoint

Der Basis-Endpoint lautet:

```
https://SUB-DOMAIN/trck/ebasket/
```

SUB-DOMAIN wird individuell pro Advertiser/Kunde konfiguriert. Der Endpoint akzeptiert die GET-Parameter `json` (Pflicht) und optional `currency`.

Pflichtparameter

Parameter	Beschreibung
json	URL-encodiertes JSON-Array mit allen Basket-Items (Pflichtfeld). Ein Objekt pro bestelltem Produkt.

Optionaler Parameter

Parameter	Beschreibung
currency	Übergabe der Währung. Standard: EUR. Bei Bedarf z. B. CHF, USD, ...

Beispielaufrufe

Mit Standard-Währung (EUR)

```
https://DOMAIN/trck/ebasket/?json=[{"campaign_id":"CAMPAIGN_ID","trigger_id":"TRIGGER_ID","token":"TOKEN","emid":"EMID","amount":"AMOUNT","price":"PRICE","article_number":"ARTICLE_NUMBER","productname":"PRODUKTNAME","category":"CATEGORY","additional":{"VARIABLE":"VALUE","VARIABLE2":"VALUE2"}}]&currency=EUR
```

Mit alternativer Währung (CHF)

```
https://DOMAIN/trck/ebasket/?json=[{"campaign_id":"1","trigger_id":"5","token":"12345","emid":"abc123","amount":"2","price":"19.90","article_number":"SKU-123","productname":"Sneaker XY","category":"Schuhe"}]&currency=CHF
```

Anforderungen an den json-Parameter

- **URL-Encoding:** Der komplette `json`-Parameter muss URL-encodiert übertragen werden, damit Sonderzeichen korrekt verarbeitet werden.
- **Struktur:** Das JSON ist ein Array von Objekten (ein Objekt pro Basket-Item).
- **Pflichtfelder je Item:** `campaign_id`, `trigger_id`, `token`, `action_id` (bzw. `emid` je nach Implementierung), `amount`, `price`, `article_number`, `productname`, `category`.
- **Additional-Daten:** Über das Feld `additional` können beliebige Key-Value-Paare (z. B. `vouchercode`, `customerType`, ...) übergeben werden.

Kurzreferenz JSON-Item

```
{
  "campaign_id": "CAMPAIGN_ID",
  "trigger_id": "TRIGGER_ID",
  "token": "TOKEN",
  "action_id": "EMID",
  "amount": "AMOUNT",
  "price": "PRICE",
  "article_number": "ARTICLE_NUMBER",
  "productname": "PRODUCT_NAME",
  "category": "CATEGORY",
  "additional": {
    "vouchercode": "WINTER20",
    "zusatzinfo": "additional_1"
  }
}
```

JSON-Struktur

JSON-Struktur

Beim Baskettracking wird der Warenkorb als **JSON-Array** im GET-Parameter `json` an den Endpoint übergeben.

Jedes bestellte Produkt wird dabei als eigenes **Objekt** innerhalb des Arrays übermittelt.

Beispiel-JSON

```
[
  {
    "campaign_id": "CAMPAIGN_ID",
    "trigger_id": "TRIGGER_ID",
    "token": "TOKEN",
    "action_id": "EMID",
    "amount": "AMOUNT",
    "price": "PRICE",
    "article_number": "ARTICLE_NUMBER",
    "productname": "PRODUCT_NAME",
    "category": "CATEGORY",
    "additional": {
      "vc": "WINTER20",
      "zusatzinfo": "additional_1"
    }
  }
]
```

Variablenbeschreibung Basket-Item

Variable	Pflichtfeld	Beschreibung
<code>campaign_id</code>	<input type="checkbox"/>	Kampagnen-ID
<code>trigger_id</code>	<input type="checkbox"/>	Trigger-ID
<code>token</code>	<input type="checkbox"/>	Bestellnummer / Order ID

Variable	Pflichtfeld	Beschreibung
<code>action_id</code>	☐	EMID (eindeutige Nutzer-ID, die über den Clicklink übergeben wird)
<code>amount</code>	☐	Anzahl, wie oft das Produkt gekauft wurde
<code>price</code>	☐	Netto-Preis des einzelnen Produkts (Dezimaltrennzeichen = Punkt <code>.</code>)
<code>article_number</code>	☐	Produkt-ID
<code>productname</code>	☐	Produktname
<code>category</code>	☐	Produktkategorie
<code>additional</code>	☐	Optionales Key-Value-Array mit weiteren Informationen (z. B. Gutscheincode, Kundentyp, Zusatzinfos)

Variablenbeschreibung Order-Ebene

Variable	Pflichtfeld	Beschreibung
<code>currency</code>	☐	Übergabe der Währung. Standard: EUR. Bei Bedarf kann CHF, USD, ... gesetzt werden.

Hinweise zur Struktur

- Das JSON ist immer ein **Array von Objekten**.
- Jedes Objekt entspricht einem **bestellten Produkt** (Basket-Item).
- Alle mit ☐ markierten Felder sind **Pflichtfelder**. Ohne diese kann die Transaktion nicht korrekt verarbeitet werden.
- Der Wert `action_id` (früher auch als `emid` bezeichnet) ist entscheidend für die **Attribution der Customer Journey**.
- Über `additional` lassen sich flexible, kundenspezifische Daten übergeben, z. B.:
 - Gutscheincode
 - Kundentyp (Neu-/Bestandskunde)
 - Affiliate-spezifische Zusatzinfos

Beispiel mit mehreren Artikeln

```
[
  {
    "campaign_id": "1",
    "trigger_id": "5",
    "token": "12345",
    "action_id": "abc123",
    "amount": "1",
    "price": "49.90",
    "article_number": "SKU-001",
    "productname": "Sneaker XY",
    "category": "Schuhe",
    "additional": {
      "vc": "WINTER20",
      "customerType": "Neukunde"
    }
  },
  {
    "campaign_id": "1",
    "trigger_id": "5",
    "token": "12345",
    "action_id": "abc123",
    "amount": "2",
    "price": "19.90",
    "article_number": "SKU-002",
    "productname": "Socken 2er Pack",
    "category": "Accessoires"
  }
]
```

Implementierung

PHP Implementierung

Dieses Beispiel zeigt, wie der Basket serverseitig in PHP aufgebaut, sauber **URL-encodiert** und an den Endpoint gesendet wird.

Es werden zwei Varianten gezeigt: **cURL** (empfohlen) und `file_get_contents` (einfach).

Beispiel mit cURL (empfohlen)

```
<?php
// Beispielhafte Datenquellen (Shop-Objekt / eigene Funktionen)
$positions = $Shopsystem->getOrderedPositions(); // Array mit bestellten Positionen
$emid      = $Shopsystem->getEmid();             // EMID (vom Click übergeben & serverseitig
gespeichert)
$orderId   = $Shopsystem->ordernumber;          // Eindeutige Order-ID

$basket = [];

foreach ($positions as $pos) {
    $basket[] = [
        'campaign_id' => (string) $pos['campaign_id'], // IDs als String halten
        'trigger_id'  => (string) $pos['trigger_id'],
        'token'       => (string) $orderId,
        'action_id'   => (string) $emid,              // EMID für Attribution
        'amount'      => (string) $pos['amount'],     // Menge je Position
        'price'       => number_format((float) $pos['net_price'], 2, '.', ''), //
Netto/Stück
        'article_number' => (string) $pos['sku'],
        'productname'   => (string) $pos['name'],
        'category'     => (string) $pos['category'],
        'additional'   => array_filter([
            'vouchercode' => $pos['voucher'] ?? null,
            'customerType' => $pos['customerType'] ?? null,
        ], static fn($v) => $v !== null && $v !== '')
    ];
}
```

```

$endpoint = 'https://SUB-DOMAIN/trck/ebasket/';

// GET-Parameter aufbauen (json wird automatisch URL-encodiert)
$query = http_build_query([
    'json'      => json_encode($basket, JSON_UNESCAPED_UNICODE | JSON_UNESCAPED_SLASHES),
    'currency' => 'EUR', // optional, Standard EUR
]);

$url = $endpoint . '?' . $query;

// cURL-Request (GET)
$ch = curl_init();
curl_setopt_array($ch, [
    CURLOPT_URL           => $url,
    CURLOPT_RETURNTRANSFER => true,
    CURLOPT_FOLLOWLOCATION => true,
    CURLOPT_TIMEOUT       => 10,
    CURLOPT_SSL_VERIFYPEER => true,
    CURLOPT_SSL_VERIFYHOST => 2,
    // Optional: Logging eines Response-Headers
    CURLOPT_HEADER        => false,
]);

$response = curl_exec($ch);
$httpCode = curl_getinfo($ch, CURLINFO_HTTP_CODE);

if ($response === false || $httpCode >= 400) {
    error_log('Basket-Tracking fehlgeschlagen: HTTP ' . $httpCode . ' - ' . curl_error($ch));
    // Optional: Retry-Logik / Queueing
}

curl_close($ch);

```

Alternative mit `file_get_contents` (einfach)

```
<?php
$endpoint = 'https://SUB-DOMAIN/trck/ebasket/';
$query = http_build_query([
    'json' => json_encode($basket, JSON_UNESCAPED_UNICODE | JSON_UNESCAPED_SLASHES),
    'currency' => 'EUR',
]);

$url = $endpoint . '?' . $query;

$context = stream_context_create([
    'http' => [
        'method' => 'GET',
        'timeout' => 10,
        'header' => [
            'Accept: application/json',
        ]
    ],
    'ssl' => [
        'verify_peer' => true,
        'verify_peer_name' => true,
    ],
]);

$result = @file_get_contents($url, false, $context);

if ($result === false) {
    error_log('Basket-Tracking fehlgeschlagen (file_get_contents). URL: ' . $url);
}
```

Voucher

Neue Seite

Es ist möglich, eine Transaktion mit einem Voucher über einen Serversided Trackingaufruf in das Partnerprogramm einzuspielen. Dafür stellt das System von easy.MARKETING einen Aufruf zur Verfügung, welcher mit entsprechenden Daten befüllt werden muss. Da es sich um ein serverseitiges Tracking handelt, muss die trackingrelevante ID (emid) des Nutzers vom Server des Advertisers an das System zurückgespielt werden.

Der Voucher wird als Array im JSON-Format im GET-Parameter "json=" an den Endpoint. Dazu stellen wir im unteren Bereich der Dokumentation ein Beispiel-Snippet in PHP zur Verfügung.

Wichtig ist, dass das Voucher-Array pro bestellter Position mit den Daten der Position befüllt wird.

Endpoint

`https://SUB-DOMAIN/trck/ebasket/`

Beispiel-Trackingaufruf

```
https://SUB-  
DOMAIN*/trck/ebasket/?json=[{"campaign_id":"CAMPAIGN_ID","trigger_id":"TRIGGER_ID","token":"TO  
KEN","emid":"EMID","amount":"AMOUNT","price":"PRICE","article_number":"ARTICLE_NUMBER","produc  
tname":"PRODUKTNAME","category":"CATEGORY","additional":{"VARIABLE":"VALUE","VARIABLE2":"VALUE  
2"}}, {"campaign_id":"CAMPAIGN_ID","trigger_id":"TRIGGER_ID","token":  
"TOKEN","emid":"","amount":"AMOUNT","price":"DISCOUNTPRICE","article_number":"VOUCHER_NUMBER",  
"productname":"VOUCHER_NAME","category":"VOUCHER_CATEGORY","additional":{"VARIABLE":"VALUE","V  
ARIABLE2":"VALUE2"}}]
```

JSON-Aufbau

```
[  
  {  
    "campaign_id": "CAMPAIGN_ID",
```

```
"token": "TOKEN",
"trigger_id": "TRIGGER_ID",
"action_id": "EMID",
"amount": "AMOUNT",
"price": "DISCOUNTPRICE",
"article_number": "VOUCHER_NUMBER",
"productname": "VOUCHER_NAME",
"category": "VOUCHER_CATEGORY",
"additional": {
  "VARIABLE": "VALUE",
  "VARIABLE2": "VALUE2"
}
}
```

Exklusives Gutscheintracking

Wenn das exklusive Gutscheintracking genutzt werden soll, muss der Gutschein zusätzlich als GET-Parameter vc nach dem json übergeben werden.

JSON-Aufbau

```
[
  {
    "campaign_id": "CAMPAIGN_ID",
    "token": "TOKEN",
    "trigger_id": "TRIGGER_ID",
    "action_id": "EMID",
    "amount": "AMOUNT",
    "price": "DISCOUNTPRICE",
    "article_number": "VOUCHER_NUMBER",
    "productname": "VOUCHER_NAME",
    "category": "VOUCHER_CATEGORY",
    "additional": {
      "vc": "VOUCHER_NUMBER",
      "VARIABLE1": "VALUE1",
      "VARIABLE2": "VALUE2"
    }
  }
]
```

Umsetzung im Falle von campaign_alias

Es ist möglich, eine Transaktion mit Basket über einen Serversided Trackingaufruf in das Partnerprogramm einzuspielen. Dafür stellt das System von easy.MARKETING einen Aufruf zur Verfügung, welcher mit entsprechenden Daten befüllt werden muss. Da es sich um ein serverseitiges Tracking handelt, muss die trackingrelevante ID des Nutzers (emid) vom Server des Advertisers an das System zurückgespielt werden.

Der Basket wird als Array im JSON-Format im GET-Parameter "json=" an den Endpoint. Dazu stellen wir im unteren Bereich der Dokumentation ein Beispiel-Snippet in PHP zur Verfügung.

Wichtig ist, dass das Basket-Array pro bestellter Position mit den Daten der Position befüllt wird.

Endpoint

`https://SUB-DOMAIN/trck/ebasket/`

Beispiel-Trackingaufruf

```
https://DOMAIN*/trck/ebasket/?json=[{"campaign_alias":"CAMPAIGN_ALIAS","trigger_id":"TRIGGER_ID","token":"TOKEN","emid":"EMID","amount":"AMOUNT","price":"PRICE","article_number":"ARTICLE_NUMBER","productname":"PRODUKTNAME","category":"CATEGORY","additional":{"VARIABLE":"VALUE","VARIABLE2":"VALUE2"}}]
```

JSON-Aufbau

```
[
  {
    "campaign_alias": "CAMPAIGN_ALIAS",
    "token": "TOKEN",
    "trigger_id": "TRIGGER_ID",
    "action_id": "EMID",
```

```

"amount": "AMOUNT",
"price": "PRICE",
"article_number": "ARTICLE_NUMBER",
"productname": "PRODUCT_NAME",
"category": "CATEGORY",
"additional": {
  "vc": "WINTER20",
  "zusatzinfo": "additional_1"
}
}
]

```

Variablenbeschreibung

Variable	Pflichtfeld	Beschreibung
campaign_alias	<input type="checkbox"/>	Kampagnen-Alias
trigger_id	<input type="checkbox"/>	Trigger-ID
token	<input type="checkbox"/>	Bestellnummer / Order ID
action_id	<input type="checkbox"/>	EMID (eindeutige Nutzer-ID, die über den Clicklink übergeben wird)
amount	<input type="checkbox"/>	Anzahl, wie oft das Produkt gekauft wurde
price	<input type="checkbox"/>	Netto-Preis des einzelnen Produkts (Dezimaltrennzeichen = Punkt <input type="text" value="."/>)
article_number	<input type="checkbox"/>	Produkt-ID
productname	<input type="checkbox"/>	Produktname
category	<input type="checkbox"/>	Produktkategorie
additional	<input type="checkbox"/>	Optionales Key-Value-Array mit weiteren Informationen (z. B. Gutscheincode, Kundentyp, Zusatzinfos)

Gutscheinimplementierung

Bei vielen Shops werden den Usern Gutscheine angeboten um Rabatte zu gewähren. Oftmals werden diese Gutscheine mit in die Berechnung von Vergütungen eingerechnet.

Rabattgutscheine:

Für Rabattgutscheine wird ein zusätzliches Basketitem innerhalb des JSON-Parametes hinzugefügt. Der abzuziehende Betrag wird als negativer turnover hinterlegt.

Beispiel-Trackingaufruf mit Gutscheinitem

```
https://SUB-DOMAIN*/trck/ebasket/?json=[{"campaign_alias":"CAMPAIGN_ALIAS","trigger_id":"TRIGGER_ID","token":"TOKEN","action_id":"EMID","amount":"AMOUNT","price":"PRICE","article_number":"ARTICLE_NUMBER","productname":"PRODUKTNAME","category":"CATEGORY","additional":{"VARIABLE":"VALUE","VARIABLE2":"VALUE2"}}, {"campaign_alilas":"CAMPAIGN_ALIAS","trigger_id":"TRIGGER_ID","token":"TOKEN","emid":"","amount":"AMOUNT","price":"DISCOUNTPRICE","article_number":"VOUCHER_NUMBER","productname":"VOUCHER_NAME","category":"VOUCHER_CATEGORY","additional":{"VARIABLE":"VALUE","VARIABLE2":"VALUE2"}}]
```

JSON-Aufbau eines Gutscheinitems

```
[
  {
    "campaign_alias": "CAMPAIGN_ALIAS",
    "token": "TOKEN",
    "trigger_id": "TRIGGER_ID",
    "action_id": "EMID",
    "amount": "AMOUNT",
    "price": "DISCOUNTPRICE",
    "article_number": "VOUCHER_NUMBER",
    "productname": "VOUCHER_NAME",
    "category": "VOUCHER_CATEGORY",
    "additional": {
      "VARIABLE": "VALUE",
      "VARIABLE2": "VALUE2"
    }
  }
]
```

Implementierung in PHP

Code-Beispiel in PHP, welches auf der Bestellabschlussseite integriert werden kann:

```

<?php
$basket = [];
$positions = $Shopsystem->getOrderedPositions(); // Alle bestellten Positionen aus dem Shop-System
$emid = $Shopsystem->getEmid(); // EMID aus dem Shop-System
foreach ($positions as $position) {
    // für jede bestellte Position
    $basketItem = [
        'campaign_alias' => Kampagnen-Name,
        'trigger_id' => 1,
        'token' => $Shopsystem->ordernumber, // Bestellnummer aus dem Shopsystem
        'action_id' => $emid, // EMID, welche vom easy.MARKETING-System im Click übergeben wurde
        'amount' => $Shopsystem->positionAmount, // Bestellte Anzahl der Position
        'price' => $Shopsystem->positionPrice, // Preis der bestellten Position
        'article_number' => $Shopsystem->articleNumber, // Artikelnummer der bestellten Position
        'productname' => $Shopsystem->productName, // Produktname der bestellten Position
        'category' => $Shopsystem->category, // Kategorie der bestellten Position
        'additional' => [
            'vc' => $Shopsystem->vouchercode, // Beispiel: Verwendeter Gutscheincode bei der
Bestellung
            'customerType' => $Shopsystem->customerType, // Custom-Values, welche frei befüllbar
sind
            // ....
        ]
    ];
    $basket[] = $basketItem;
}
// Aufbau der Tracking-URL
$endpoint = 'https://DOMAIN/trck/ebasket/';
$getParameter = [
    'json' => json_encode($basket),
];
$trackingUrl = $endpoint . '?' . http_build_query($getParameter);
// Aufruf der Tracking-URL via wget; hier kann auch jeder andere Client genutzt werden.
exec('wget -O /dev/null"' . $trackingUrl . '"');
Wenn andere currency als EUR verwendet wird, kann diese als getParameter eingefügt werden.
Beispiel:
    'json' => json_encode($basket),
    'currency' => 'CHF'

```

Dabei entspricht das Object \$Shopsystem dem Shop des Advertisers, welcher die Werte bereitstellt.