Umgang mit Additional Parameters auf Transaktionsebene

Diese Dokumentation erläutert die Logik der Datenverarbeitung von Zusatzparametern (`Additional Parameters`) und deren Auswirkungen auf das Tracking und externe Systeme (z. B. Tagmanager).

Verarbeitungslogik

Obwohl die Funktion `eamTrckAddBasketItem()` Parameter für das einzelne Produkt entgegennimmt, werden diese Werte in unserem System auch an den **Gesamtdatensatz der Transaktion** (Order-Ebene) geschrieben.

Da eine Transaktion aus mehreren Artikeln bestehen kann, gilt bei der Verarbeitung folgende Regel:

> **Der letzte Wert gewinnt.** Wenn unterschiedliche Artikel denselben Parameter-Key (z. B. `vc` oder `shipping_type`) mit unterschiedlichen Werten übergeben, wird der Wert des **letzten** `eamTrckAddBasketItem()`-Aufrufs als finaler Wert für die gesamte Transaktion gespeichert.

Wichtige Richtlinie für die Implementierung

Damit Daten auf Transaktionsebene (z. B. für die Nutzung als Trigger-Bedingung im Tag Manager) zuverlässig zur Verfügung stehen, muss folgendes beachtet werden:

Parameter, die für die gesamte Bestellung gelten, müssen in jedem Basket-Item mit identischem Wert übergeben werden.

Warum ist das wichtig?

1. **Tag Manager Bedingungen:** Viele Tag Manager nutzen Variablen der Transaktionsebene, um zu entscheiden, ob ein Pixel gefeuert werden soll. Ist der Wert inkonsistent oder fehlt er im letzten Artikel, schlägt die Bedingung fehl.
2. **Datenkonsistenz:** In Exporten und Reports wird der Parameter der Transaktion zugeordnet. Unterschiedliche Werte innerhalb einer Bestellung führen hier zu Zufallsergebnissen (basierend auf der Sortierung der Artikel im Code).

Beispiel: Richtige vs. Falsche Implementierung

❌ Falsch: Inkonsistente Werte

Hier würde der Wert für `customer_group` auf Transaktionsebene mit "Bestandskunde" überschrieben werden, obwohl der erste Artikel "Neukunde" meldet.

```
// 1. Artikel
eamTrckAddBasketItem(..., { 'customer_group': 'Neukunde' });

// 2. Artikel
eamTrckAddBasketItem(..., { 'customer_group': 'Bestandskunde' });

// Ergebnis Transaktionsebene: 'Bestandskunde'
```

✅ Richtig: Konsistente Werte

Soll ein Parameter sicher für die gesamte Transaktion gewertet werden, muss er bei jedem Aufruf identisch sein.

```
// 1. Artikel
eamTrckAddBasketItem(..., {
```

```
  'vc': 'SOMMER24',
  'shipping_method': 'Express',
  'color': 'Rot'
});

// 2. Artikel
eamTrckAddBasketItem(..., {
  'vc': 'SOMMER24',
  'shipping_method': 'Express',
  'color': 'Blau'
});

// Ergebnis Transaktionsebene: 'SOMMER24' und 'Express' sind korrekt gesetzt.
```

Zusammenfassung für Entwickler

1. **Globale Parameter** (Gutscheincodes, Kundengruppen, Versandarten) immer an **alle** Artikel der Bestellung hängen.
2. Sicherstellen, dass die Werte innerhalb eines Warenkorbs für diese globalen Keys **identisch** sind.
3. Nur **produktspezifische Parameter** (Größe, Farbe) dürfen variieren, wobei zu beachten ist, dass im Transaktions-Header nur der Wert des letzten Artikels sichtbar sein wird.

Revision #1
Created 2026-04-02 07:24:20 UTC by Daniel Hörenkamp
Updated 2026-04-02 07:25:20 UTC by Daniel Hörenkamp