

---

# **ItkWebGains**

Anwenderdokumentation

André Fischer

30.03.2026

## Inhaltsverzeichnis

<b>1 Was macht dieses Plugin?</b>	<b>2</b>
1.1 Diagramm: Partner bis Auswertung . . . . .	3
<b>2 Wo du die Einstellungen findest</b>	<b>4</b>
<b>3 Typischer Ablauf nach der Installation</b>	<b>4</b>
<b>4 Plugin-Konfiguration (technisch, alle Felder)</b>	<b>4</b>
4.1 Karte „WebGains Grundeinstellungen“ . . . . .	4
4.2 Tracking Einstellungen . . . . .	5
4.3 Export Einstellungen . . . . .	5
4.4 Clickthrough Parameter . . . . .	6
<b>5 Produktfeed-URL für WebGains</b>	<b>6</b>
<b>6 Admin-Exporte (CSV)</b>	<b>7</b>
<b>7 Daten an der Bestellung (Clickthrough)</b>	<b>7</b>
<b>8 Grenzen und was das Plugin nicht ist</b>	<b>8</b>
<b>9 Troubleshooting</b>	<b>8</b>
<b>10 Kurz für die Übergabe an Marketing</b>	<b>9</b>

## 1 Was macht dieses Plugin?

**WebGains** ist ein Affiliate-Netzwerk: Partner verlinken auf den Shop und erhalten Provision, wenn über diese Zuordnung ein Kauf zustande kommt. Das Plugin **ItkWebGains** ist die technische Anbindung in Shopware 6.6 und erledigt für dich im Wesentlichen drei Dinge:

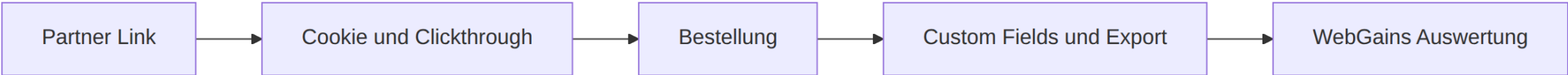
1. **Tracking im Storefront** – Es können Skripte (Container / Landing) eingebunden und – sofern aktiviert – **Clickthrough-Daten** über Parameter wie **siteid** und **utm\_source** in einem Cookie gemerkt werden. Bei der Bestellung werden diese Infos an der **Bestellung** hinterlegt (Custom Fields), damit spätere Auswertungen und Exporte stimmen.
2. **Transaktions-Tracking** – Auf der **Bestellabschlussseite** wird – wenn du es einschaltest und die DSGVO-Logik greift – ein **Transaktions-Pixel** bzw. die dafür vorgesehene Ausspielung genutzt, damit WebGains den Umsatz der zugeordneten Partneraktion sehen kann.
3. **Produktfeed** – Unter einer festen Storefront-URL wird eine **CSV-Datei** mit Produktdaten bereitgestellt, die du bei WebGains als Katalog hinterlegen kannst. Optional wird der Feed **gecached**, damit große Kataloge den Server nicht bei jedem Abruf neu berechnen.

Zusätzlich gibt es im **Admin** (API-Routen) **CSV-Exporte** für Transaktionen (neu, Approve, Reject, Adjust) und den Produktfeed – typischerweise angebunden über ein Admin-Modul oder externe Tools, die die Shopware-Admin-API nutzen.

**Voraussetzungen:** Aktives **WebGains-Programm** mit **Program ID** und **Event ID** (und ggf. weiteren IDs aus deinem WebGains-Account). **Kein** separates Composer-Paket nötig – nur dieses Plugin und ein funktionierender Shop.

---

**1.1 Diagramm: Partner bis Auswertung**



## 2 Wo du die Einstellungen findest

**Erweiterungen → Meine Erweiterungen → ITK WebGains Affiliate Marketing → Konfiguration.**

Oben kannst du den **Sales Channel** wählen. WebGains-Konfiguration ist **kanalspezifisch**: Jeder Kanal kann eigene IDs, eigenes Tracking und einen eigenen Feed haben.

---

## 3 Typischer Ablauf nach der Installation

1. In der Konfig **Plugin aktivieren** (**active** = an) für den gewünschten Kanal.
  2. **Program ID, Event ID, Container ID** (laut WebGains-Dokumentation) eintragen.
  3. Entscheiden: **Container Tag** und **Transaction Tracking** an oder aus.
  4. **DSGVO**: Wenn du Consent brauchst, „**GDPR berücksichtigen**“ anlassen und den **Standard-Wert** setzen (siehe technischer Abschnitt unten).
  5. **Produktfeed**: aktivieren, URL bei WebGains eintragen (<https://deine-domain/webgains/feed/products.csv>).
  6. **Testbestellung** mit Partnerlink (Cookie gesetzt) durchspielen und in WebGains prüfen, ob die Zuordnung ankommt.
  7. Logs bei Problemen: [var/log/webgains.log](#).
- 

## 4 Plugin-Konfiguration (technisch, alle Felder)

Alle Keys hängen am Präfix `ItkWebGains.config`. (im Admin siehst du nur die Felder).

### 4.1 Karte „WebGains Grundeinstellungen“

Feldname (intern)	Typ	Bedeutung
<b>active</b>	bool	Hauptschalter. Wenn aus, liefert der Feed 404 und Tracking-Skripte werden nicht eingebunden.
<b>programId</b>	Text	WebGains Program ID.
<b>eventId</b>	Text	WebGains Event ID (für Transaktions-Tracking).
<b>containerId</b>	Text	Container ID für den WebGains Tag Manager / Skript-Container.
<b>campaignId</b>	Text	Optionale Campaign ID.

## 4.2 Tracking Einstellungen

Feldname	Typ	Bedeutung
<b>enableContainerTag</b>	bool	Steuert, ob Container-Skripte auf den Storefront-Seiten erzeugt werden (Default laut XML: an).
<b>enableTransactionTracking</b>	bool	Transaktions-Tracking auf der Bestellbestätigungsseite (Default: an).
<b>respectGdpr</b>	bool	Wenn an, wird Tracking nur ausgeführt, wenn laut Implementierung Consent gegeben ist (Default: an).
<b>gdprDefault</b>	Auswahl	Wert <b>-1</b> = unbekannt, <b>0</b> = DSGVO gilt nicht, <b>1</b> = DSGVO gilt – wird in der Tracking-Logik ausgewertet, wenn kein expliziter Consent-Zustand ermittelt werden kann.

## 4.3 Export Einstellungen

Feldname	Typ	Bedeutung
<b>enableProductFeed</b>	bool	Öffentlichen CSV-Feed unter <code>/webgains/feed/products.csv</code> freischalten.

---

Feldname	Typ	Bedeutung
<b>enableProductFeedCache</b>	bool	Feed über Symfony Cache ( <code>cache . app</code> ) zwischenspeichern.
<b>productFeedCacheLifetime</b>	int	Cache-Laufzeit in <b>Minuten</b> (Default 60).
<b>enableTransactionExport</b>	bool	Schaltet die Nutzbarkeit der Transaktions-Export-Logik frei (für Admin-API; fachlich mit internen Prozessen abstimmen).
<b>transactionValidationDays</b>	int	Zeitraum in Tagen für Validierungs-Kontexte im Export (Default 30).

---

#### 4.4 Clickthrough Parameter

---

Feldname	Typ	Bedeutung
<b>trackClickthrough</b>	bool	Ob <code>utm_source</code> / <code>siteid</code> (und verwandte Daten) im Cookie <code>wg_clickthrough</code> gespeichert werden.
<b>clickthroughCookieLifetime</b>	int	Cookie-Lebensdauer in <b>Tagen</b> (Default 30).

---

## 5 Produktfeed-URL für WebGains

Öffentlicher Endpunkt (Storefront):

```
1 GET https://<deine-shop-domain>/webgains/feed/products.csv
```

Voraussetzungen: Plugin **aktiv**, **Product Feed aktiv**, Aufruf im Kontext des richtigen **Sales Channels** (Domain muss zum Kanal passen). Sonst antwortet der Server mit **404 Not Found**.

Der Feed wird mit **Content-Disposition: attachment** ausgeliefert – für WebGains meist unkritisch, Hauptsache die URL ist erreichbar. Die **Cache-Dauer** steuert sowohl den internen Cache (falls an) als auch den HTTP-Header **Cache-Control: max-age=...** (in Sekunden, aus den Minuten der Config umgerechnet, mindestens sinnvoller Default).

## 6 Admin-Exporte (CSV)

Über die **Admin-API** (nicht im klassischen Storefront-Menü) stellt das Plugin u. a. bereit:

---

Methode	Pfad	Zweck (kurz)
POST	/api/_action/ webgains/export/ transactions/new	Neue Transaktionen im Zeitraum ( <b>from, to</b> , optional <b>salesChannelId</b> ).
POST	/api/_action/ webgains/export/ transactions/ approve	Approve-CSV anhand Bestellnummern.
POST	/api/_action/ webgains/export/ transactions/reject	Reject-CSV.
POST	/api/_action/ webgains/export/ transactions/adjust	Adjust-CSV.
POST	/api/_action/ webgains/export/ products	Produktfeed wie Storefront, aber als Download aus dem Admin-Kontext.

---

Du brauchst ein **authentifiziertes** Admin-Request (OAuth / Integrations-Account), genaue Request-Bodies stehen in der Entwicklerdoku bzw. im PHP-Controller.

---

## 7 Daten an der Bestellung (Clickthrough)

Wenn ein Cookie `wg_clickthrough` vorhanden ist und eine Bestellung geschrieben wird, versucht das Plugin, folgende **Order Custom Fields** zu setzen (sofern die Werte nicht leer sind):

- `webgains_clickthrough_siteid`
- `webgains_clickthrough_clickref`
- `webgains_clickthrough_utm_source`
- `webgains_clickthrough_timestamp`

Diese Felder müssen in Shopware als **Custom Fields** an der **Order** existieren und für den Editor sichtbar sein – sonst schlagen Updates fehl oder Werte erscheinen nicht. Wenn ihr das Set noch nicht angelegt habt, muss das einmalig im Projekt erfolgt werden (Migration oder Admin).

---

## 8 Grenzen und was das Plugin nicht ist

- **Kein** Ersatz für eure **Rechts- oder Cookie-Beratung** – **respectGdpr** und Consent hängen mit Cookie-Tool und Theme zusammen.
- **Keine** Garantie, dass jeder Browser oder jede Adblock-Erweiterung Skripte oder Pixel durchlässt.
- **Provision und Attribution** werden letztlich von **WebGains** berechnet; das Plugin liefert nur Daten und Schnittstellen.

---

## 9 Troubleshooting

Problem	Prüfpunkt
Kein Tracking / kein Pixel	<b>active, enableTransactionTracking</b> , DSGVO-Consent, Adblocker, webgains.log
Feed 404	<b>active, enableProductFeed</b> , richtige Domain = richtiger Sales Channel
Keine Attribution	Cookie wg_clickthrough gesetzt? <b>trackClickthrough</b> an? Laufzeit nicht abgelaufen?
Custom Fields fehlen an der Order	Felddefinitionen in Shopware für <b>order</b> angelegt?
Alte Feed-Daten	Cache: <b>enableProductFeedCache</b> + Lifetime; Shopware-Cache leeren

## 10 Kurz für die Übergabe an Marketing

- IDs kommen aus dem **WebGains-Account**.
- **Feed-URL** exakt kopieren, in WebGains als Produktquelle eintragen.
- **Testkauf** mit echtem Partnerlink und danach Daten in WebGains prüfen.
- **DSGVO:** Nur testen, wenn Consent wie im Live-Shop erteilt wird.