

Google Ads Skripte – wertvolle Helfer für eine effiziente Konto-Betreuung

Unser SEA Experte Martin hat sich intensiv mit dem Thema Google Ads- Skripte befasst. In seinem Beitrag geht er u.a. auf die Möglichkeiten und verschiedenen Arten ein. Los geht's, seid gespannt ...

Google Ads-Skripte automatisieren Prozesse nach vorher festgelegten Regeln. Die Nutzung von Automatisierung in einem [Google Ads-Account](#) lohnt sich für sehr große Konten und bei Account-Managern, die mehrere Konten gleichzeitig betreuen. Die Mini-Programme werden entweder in festgelegten Intervallen oder manuell ausgeführt.

Google Ads-Skripte basieren auf der Programmiersprache JavaScript. Wer ein gutes Grundverständnis für Programmiersprachen hat oder zumindest gut frickeln kann, wird sich schnell in die Skriptsprache einarbeiten. Ein gutes kostenloses Einführungs-Tutorial findet man hier.

Möglichkeiten von Google Ads-Skripten

Google Ads-Skripte können sowohl auf Daten innerhalb des Google Ads-Kontos als auch auf externe Quellen zugreifen. Externe Quellen können z.B. alternative Anbieter, Wetterdaten oder auch Daten der Zielseite sein.

Der Funktionsumfang von Google Ads-Skripten ist leider begrenzt und vor allem für die Optimierung von Textanzeigen im Such- und Partnernetzwerk geeignet. Skripte für Shopping-, Display- und Videokampagnen können nur sehr wenig Performance-Optimierungs-Potentiale abdecken. Funktionen findet man für die Bearbeitung von:

- Kampagnen
- Anzeigengruppen
- Textanzeigen
- Keywords
- Filter
- Statistiken (begrenzt)
- CPC-Gebote
- und Budgets

Funktionen, die noch hilfreich wären, sind Zugriffe auf fehlende Statistiken, wie mögliche Impressionen, segmentierte Statistiken (z.B. Übereinstimmungstyp der Suchbegriffe) und noch einige weitere Möglichkeiten, die jedoch eher erfahrene Anwender betreffen.

Für die, die sich nicht in JavaScript einarbeiten wollen, gibt es auf dem Google Entwickler Portal eine kleine Auswahl an abgespeckten Skripten.

Ganz ohne den Code zumindest zu überfliegen, funktionieren diese allerdings auch nicht. Man braucht ein individuell formatiertes Google Tabellenblatt (z.B. in Drive) für den Export und muss dieses entsprechend in den Code in einem Platzhalter SPREADSHEET_URL einfügen und ggf. ein paar weitere Variablen wie die Empfänger-E-Mail anpassen.

Arten von Skripten

Es gibt 4 Arten von Skripten:

- Reports
- Bidding-Skripte
- Tools
- Alerts

Skripte für Reports sind immer sinnvoll. Sie geben zu einem festgelegten Zeitpunkt eine Übersicht zu vorher festgelegten KPIs. Die Übersicht kann man sich entweder via Mail schicken lassen oder in Google Drive abrufen. Neben klassischen Indikatoren wie Keyword-Performance, Anzeigen- oder Kampagnenleistung können auch Übersichten auf MCC-Basis und sogar Seitenladezeiten für mobile Seiten monitort werden. Ein großer Vorteil von Report-Skripten ist, dass auch ein weniger erfahrener Anwender sie austesten kann, da sie keine Auswirkungen auf das Konto haben.

Bidding-Skripte übernehmen im Best Case die Arbeit von individuell konzipierten Bid-Management-Systemen. Diese Skripte bieten das Potential Accounts effizienter zu optimieren als jeder Berater, da sie 24/7 alle vorher einprogrammierten KPIs im Blick haben und entsprechend Stellschrauben nachjustieren können.

Problem von Bidding-Skripten ist, dass egal wie komplex das Skript aufgebaut sein mag, es nicht ohne den menschlichen Account-Manager als Backup auskommen wird.

Automatisierte Programme haben nur einen begrenzten Horizont und einen eingeschränkten Funktionsumfang. Ein Bidding-Skript – was in der Regel auf Performance-Steigerung ausgerichtet ist – würde so zum Beispiel ein Keyword mit Absprungraten jenseits der 90% mit 100 Klicks und hohen CPC ohne Conversions (oder messbare Vorbereitung) wahrscheinlich pausieren (Alternative: weiter Geld verbrennen).

Ein guter Berater könnte aber feststellen, dass die Zielseite einfach nicht mehr zum Keywordset passt und hier durch Eingreifen Performance retten, die sonst verloren gegangen wäre.

Auch wird ein Skript kaum aus Suchbegriffen neue Anzeigengruppen erstellen können oder conversionstarke Anzeigentexte schreiben. Eine strategische Weiterentwicklung wird aktuell ebenfalls kein Bidding-Skript übernehmen können.

Bidding-Skripte können für erfahrene Account-Manager eine enorm wertvolle Unterstützung sein, um das Maximum für seine Kunden herauszuholen. Für den unerfahrenen Anwender hingegen besteht die Gefahr, sein Google Ads-Konto damit systematisch herunterzuwirtschaften.

Skript-Tools nehmen z. B. automatisierte Änderungen an Anzeigentexten vor oder pausieren diese auf Grundlage von externen Daten. Interessant kann das sein, wenn Anzeigen pausiert werden sollen, sobald auf der Zielseite aktuell keine Angebote mehr zu finden sind.

```

29 *
30 * @changelog
31 * - version 1.0
32 * - Released initial version.
33 */
34
35 var SPREADSHEET_URL = 'https://docs.google.com/spreadsheets/d/1c1y1Kp_CQ-MHIq_n4gYFj-3f';
36
37 function main() {
38   Logger.log('Using spreadsheet - %s.', SPREADSHEET_URL);
39   var spreadsheet = SpreadsheetApp.openByUrl(SPREADSHEET_URL);
40
41   var sheet = spreadsheet.getSheets()[0];
42   spreadsheet.getRangeByName('account_id').setValue(
43     AdWordsApp.currentAccount().getCustomerId());
44   sheet.getRange(1, 2, 1, 1).setValue('Date');
45   sheet.getRange(1, 3, 1, 1).setValue(new Date());
46   sheet.getRange(7, 1, sheet.getMaxRows() - 7, sheet.getMaxColumns()).clear();
47
48   var adGroupsIterator = AdWordsApp.adGroups()
49     .withCondition("Status = 'ENABLED'")
50     .withCondition("CampaignStatus = 'ENABLED'")
51     .forDateRange('LAST_7_DAYS')
52     .orderBy('Ctr ASC')
53     .withLimit(100)
54     .get();
55
56   var today = getDateInThePast(0);
57   var oneWeekAgo = getDateInThePast(7);
58   var twoWeeksAgo = getDateInThePast(14);
59   var threeWeeksAgo = getDateInThePast(21);
60
61   var reportRows = [];
62
63   while (adGroupsIterator.hasNext()) {
64     var adGroup = adGroupsIterator.next();
65     // Let's look at the trend of the ad group's CTR.
66     var statsThreeWeeksAgo = adGroup.getStatsFor(threeWeeksAgo, twoWeeksAgo);
67     var statsTwoWeeksAgo = adGroup.getStatsFor(twoWeeksAgo, oneWeekAgo);
68     var statsLastWeek = adGroup.getStatsFor(oneWeekAgo, today);
69
70     // Week over week, the ad group is declining - record that!
71     if (statsLastWeek.getCtr() < statsTwoWeeksAgo.getCtr() &&
72         statsTwoWeeksAgo.getCtr() < statsThreeWeeksAgo.getCtr()) {
73       reportRows.push([adGroup.getCampaign().getName(), adGroup.getName(),
74         statsLastWeek.getCtr(), statsLastWeek.getCost(),

```

Solche Skripte sind auch sinnvoll bei sich häufig ändernden Angebotspreisen, die dann über eine Variable in den Anzeigentexten angepasst werden. Die Verwendung von Skript-Tools bietet sich bei sehr speziellen Kundenanforderungen an, können in diesen Fällen aber sehr viel Zeit sparen.

Die letzte und wichtigste Gruppe von Skripten sind Alerts. Alerts alarmieren den Account-Manger über eine Mail direkt über vorher festgelegte Konto-Anomalien. Anomalien können zum Beispiel ein erheblicher Abfall von Positionen oder Conversions sein.

Ein gewissenhafter Konto-Betreuer wird in regelmäßigen Abständen die Performance seiner Schäfchen kontrollieren. Ein gewissenhafter Hirte mit Köpfchen schreibt hierfür ein Skript und weiß seine Herde 24/7 in bester Obhut.

Hier gilt aber, wie bei allen Skript-Arten, dass man Skripte immer gründlich testen muss, bevor man sich

auf sie verlässt.

Fazit

Google Ads-Skripte sind ein geniales Werkzeug, um Aufgaben zu automatisieren, Zeit zu sparen und somit noch mehr Spielraum für die strategische Weiterentwicklung von Konten zu haben.

Ein Skript kann aber natürlich immer nur so gut sein wie der Google Ads-Spezialist, der es geschrieben hat.

Die Kunst liegt hier in dem richtigen Maß der Automatisierung. Auch gute Coder werden Potentiale verschenken, wenn sie zu viel automatisieren und somit nicht alles für einen Kunden rausholen können. Wer jedoch zu wenig automatisiert, verschwendet unnötig Zeit, die anderweitig wesentlich effektiver eingesetzt werden könnte.