A Bing Ads Script for Maintaining Monthly Account Budgets

Last month, Bing Ads introduced their answer to Google Ads Scripts.

By now their scripts have gone into open beta so all advertisers worldwide will find them in the Bulk Operations section of their accounts.

At the time of the launch, I shared a sample script to report account-level Quality Score.

But it was quickly clear that I would need to come up with some nifty coding tricks to work around current limitations like the lack of support for labels, spreadsheets, or emails.

So this month I am sharing a script that you can copy-and-paste into your Bing Ads MCC account to keep managed accounts under a maximum monthly budget.

The script includes example code for workarounds to send emails and label things that the script changed so you can grab that code and reuse it in your own scripts.

The script I am sharing helps advertisers keep their costs under a monthly target budget.

By running the script daily, it can detect if a predetermined level of spend has been reached and then pause all currently active campaigns.

When it detects that the target spend has been reached, the script sends an email and stores the paused campaigns in a public data store.

When the script runs at the start of a new month, it uses that data store to see what campaigns it previously paused so it can turn these back on.

Monthly Account Budget Script for Bing Ads

Here’s what the script does:

- It fetches all campaigns for an account and totals up the spend so far this month.
- If the total spend exceeds a user-defined limit, all campaigns are paused.
- If the script is run during the first day of the month, it looks up which campaigns were paused last month and re-enables those.

Unlike Google Ads where scripts can be scheduled to run hourly, in Bing Ads, a script can be scheduled at most once per day.

That means you will need to set your maximum budget a bit lower than in Google because once you’re very close to your budget cap, you could get almost 24 extra hours of clicks before the script runs again and pauses campaigns.

As an alternative, you could schedule 24 copies of the same script to run at different hours.

However, this is not an endorsed workaround, so it could lead to policy issues if Bing Ads finds you’re abusing their system.

In other words, tread cautiously if you use this method.

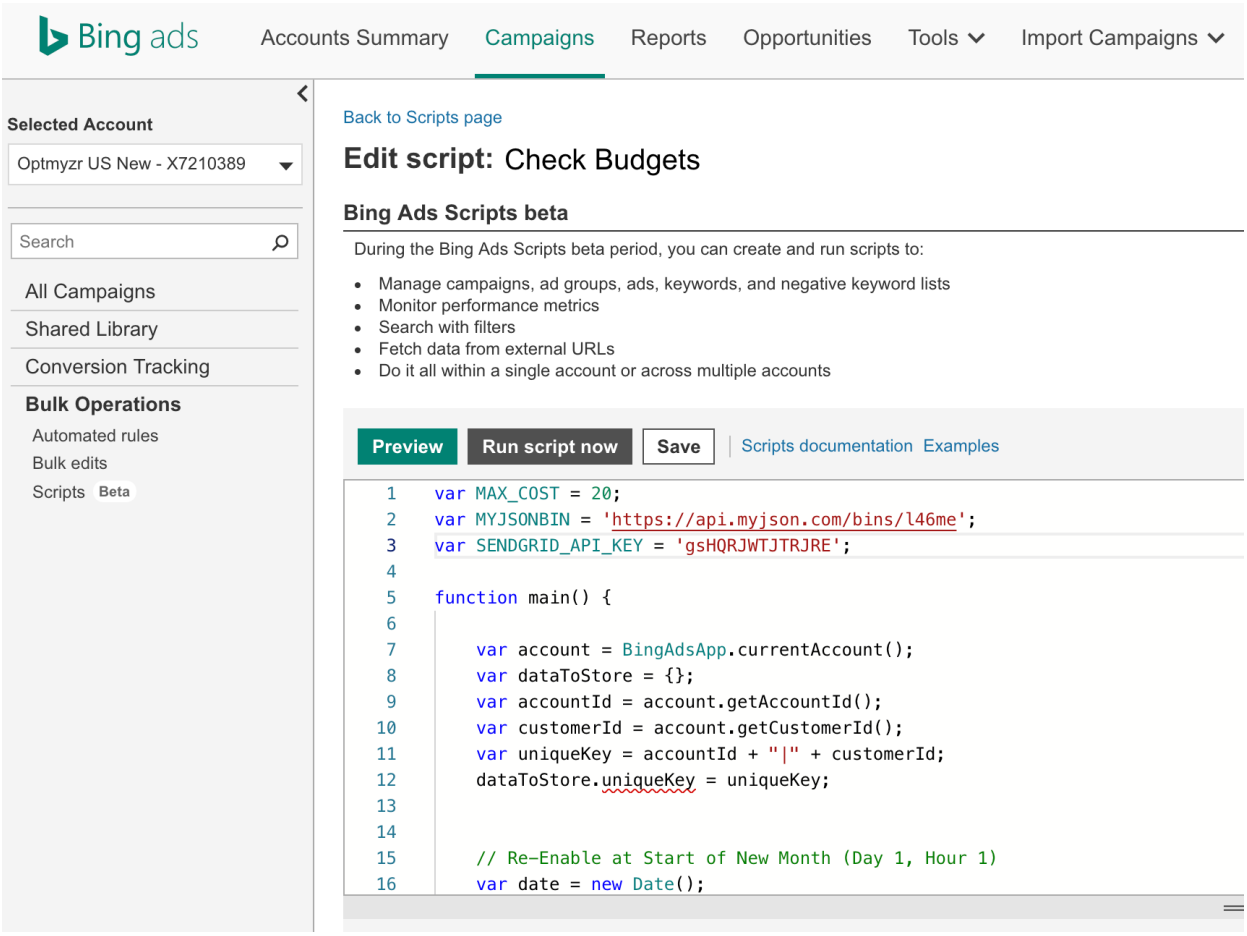
The Script

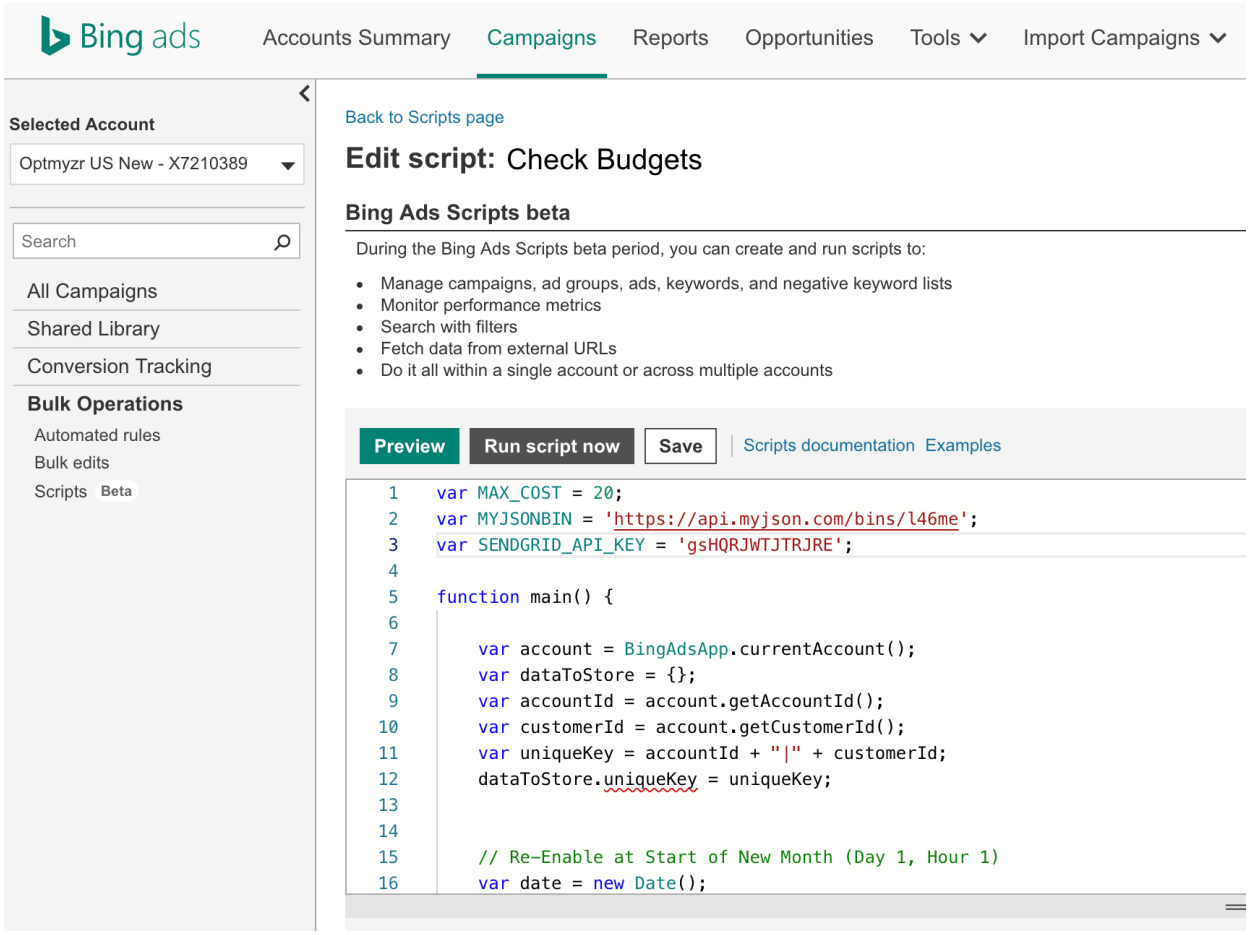
Copy this code into your Bing Ads account.

Go to Bulk Operations, then Scripts, and add this as a new script.

Replace all code of the new script with the code you copy here.

Once that’s done, go on to update the settings explained in the following paragraph.





After you copy the sample script into your Bing Ads account, it should look something like this.

Script Settings

To make this script work, you'll need to update a few settings first:

MAX_COST

This is the maximum you want an account to spend.

The script runs on your schedule, at most once per day.

So set this limit lower than your absolute maximum to give yourself some breathing room in case there are many hours between when the cost is exceeded and the next time the script runs.

MYJSONBIN

This is the URL of your own storage space for this script.

Get yours from http://myjson.com/ by saving some random text and then grabbing the resulting URL.

SENDGRID_API_KEY

This is the key to send emails through the SendGrid API.

Get yours from https://app.sendgrid.com/settings/api_keys.

EMAIL_ADDRESS

The email address that should receive a notification when campaigns are paused. The script only sends an email when changes are made to campaigns.

At this point, you're ready to test and schedule the script.

If you want to know more about how we worked around some of the limitations of scripts, the next three sections will be of interest to you.

Labels for Bing Ads Scripts

Because scripts don't yet work with labels, we created a workaround that stores information about the last time a script ran in a public JSON storage container.

Every user is supposed to create their own "bin" for every script. This is a publicly accessible URL.

To make sure the data is what we expect, we write a unique ID consisting of the Bing Ads account and Bing Ads customer ID to storage.

If this matches what the script expects, it will use the data in the storage to decide which campaigns to turn back on.

If there’s a mismatch, it will email the user a warning that they have to manually turn campaigns back on.

Email for Bing Ads Scripts

Sending emails is not natively supported in Bing Ads scripts so instead we use SendGrid’s free API tier to handle emails.

You can get your own key from this page: https://app.sendgrid.com/settings/api_keys.

Bing Ads Scripts + Google Sheets

Because many of us have been working with Google Ads Scripts for several years, we’ve become very accustomed to using Google Sheets to maintain script settings and to store reports we generate.

I was worried that Bing Ads Scripts would only work with Microsoft Excel and force us to have to use different systems to look at the reports from the different engines.

But to my surprise and delight, The Bing Ads team published a blog post with a sample script that shows how Bing Ads can interface with Google Sheets.

This is big news because it means that we can now run scripts that are nearly identical in Bing Ads and Google Ads accounts and have all the resulting output be dropped in a single Google Sheet.

Conclusion

Some of the most used scripts revolve around Quality Score, budgets, and URL checks.

Between last month’s free script and this month’s, you now have two free scripts to get started with these types of

automations for Bing Ads.

As they further refine their scripting capabilities it’ll become easier to incorporate these automations into your PPC workflows.

I look forward to covering more examples based on your feedback over the coming months.

More Resources:

Subscribe to SEJ

Get our daily newsletter from SEJ’s Founder Loren Baker about the latest news in the industry!

http://tracking.feedpress.it/link/13962/10901947