

fbRads

Facebook marketing R felhasználóknak

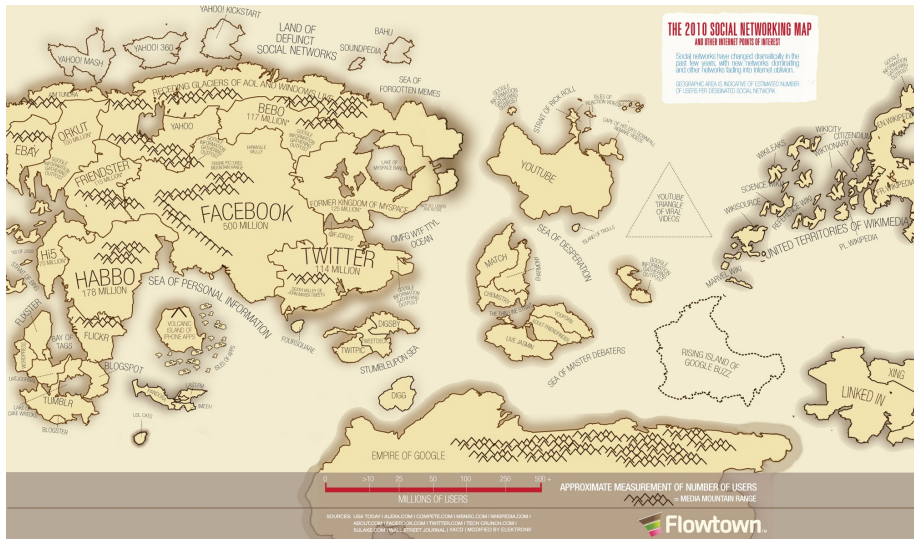
Daróczi Gergely

CARD.com

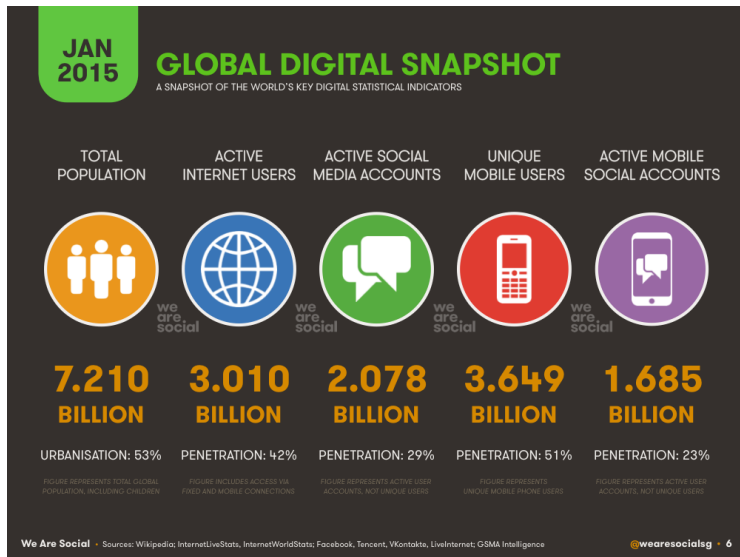
2015-09-30



When to advertise on Facebook



When to advertise on Facebook



CARD.com's view of the world



Gergely Daróczy @daroczig · Apr 11

Just received my "I ♥ R" prepaid debit card from @CARD. Will be fun to use this #rstats designed card at #user2015 :)

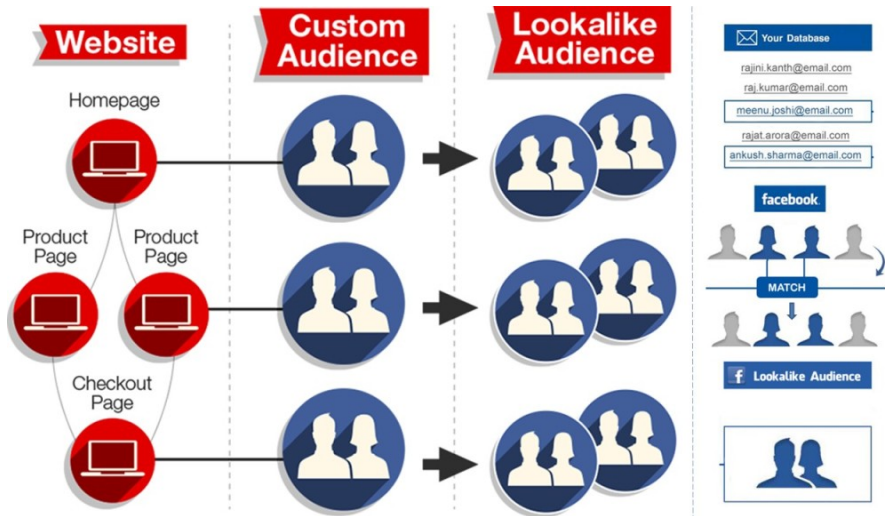


RETWEETS
10

FAVORITES
16

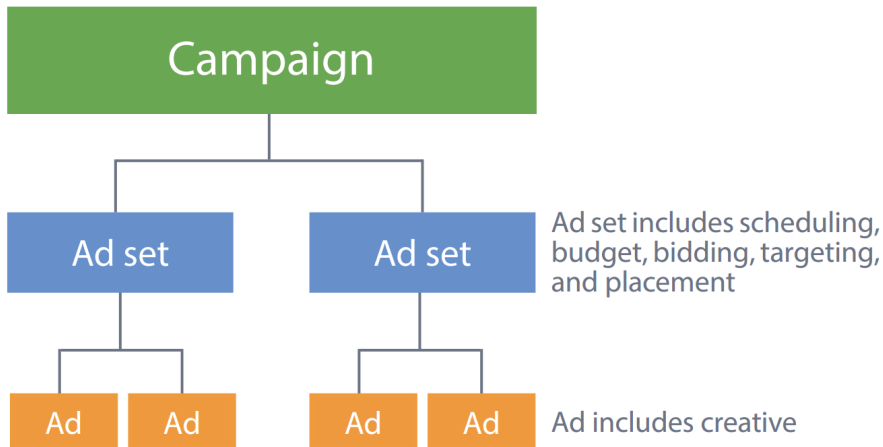


When to advertise on Facebook



Source: adparlor.com

Anatomy of a Facebook Ad Campaign



Source: Facebook Marketing API docs

Collect e-mail addresses from the R-help mailing list

Get the location of the archives:

```
> url <- 'https://stat.ethz.ch/pipermail/r-help/'
```

Get the URL of all archive files:

```
> library(RCurl)
> R.help.toc <- htmlParse(getURL(url))
> R.help.archives <- unlist(
+   xpathApply(R.help.toc, "//table//td[3]/a", xmlAttrs),
+   use.names = FALSE)
```

Download archive files:

```
> dir.create('r-help')
> for (f in R.help.archives)
+   download.file(url = paste0(url, f),
+                 file.path('help-r', f), method = 'curl')
```

Regular expression matching date format in "From" lines:

```
> dateregex <- paste('[A-Za-z]{3} [A-Za-z]{3} [0-9]{1,2} [0-9]{2}:[0-9]{2}:[0-9]{2} [0-9]{4}')
```

Extract e-mail addresses from these lines:

```
> mails <- system(paste0("zgrep -E '^From .* at .* ', dateregex, ' ./help-r/*.txt.gz"), intern = TRUE)
> mails <- sub('.*From ', '', mails)
> mails <- sub(paste0('[ ]*', dateregex, '$'), '', mails)
> mails <- sub(' at ', '@', mails)
```

Authenticate with the Facebook API



The screenshot shows the Facebook Developers console with a semi-transparent overlay containing R code and instructions. The background interface includes a top navigation bar with 'Developers', 'Products', 'Docs', 'Tools & Support', and 'News'. On the left, there's a 'Start Over' button. On the right, there's a 'Skip and Create App ID' button. The main content area of the background shows a 'Create a New App ID' dialog with fields for 'Create "Rads demo app App"', a 'Category' dropdown set to 'Business', and buttons for 'Cancel' and 'Create App ID'. The overlay text and code are as follows:

Create a token:

```
> library(httr)
> app <- oauth_app('facebook', 'your_app_id', 'your_app_secret')
> tkn <- oauth2.0_token(
+   oauth_endpoints('facebook'), app, scope = 'ads_management',
+   type = 'application/x-www-form-urlencoded')
> tkn <- tkn$credentials$access_token
```

Save this secret token (never commit to git repository) and load it in any later session:

```
> saveRDS(tkn, 'token.rds')
> tkn <- readRDS('token.rds')
```

Initialize connection to Facebook Marketing API:

```
> fbacc <- fbacc_init(fid, tkn)
```

<https://developers.facebook.com/apps/>

Create custom and lookalike audiences

Create a custom audience:

```
> aud_id <- fbad_create_audience(name = 'R-help posters',  
+   title = 'Unique e-mail addresses in R-help 1997-2015')  
  
> fbad_read_audience(audience_id = aud_id, fields = 'approximate_count')  
20
```

Seed it:

```
> fbad_add_audience(audience_id = aud_id, schema = 'EMAIL', hashes = mails)
```

Wait an hour or so:

```
> fbad_read_audience(audience_id = aud_id, fields = 'approximate_count')  
8700
```

Then create a lookalike audience based on that:

```
> fbad_create_lookalike_audience(origin_audience_id = aud_id, ratio = 0.01, country = 'HU')  
  
> fbad_read_audience(audience_id = audience, fields = 'approximate_count')  
61500
```

Create an Ad Campaign and Ad Set

Create the container:

```
> campaign <- fb_ad_create_campaign(name = 'Promoting my BURN lightning talk')
```

Define the target:

```
> target <- list(name = 'HUN R-help posters', id = aud_id)
```

Deploy the budget:

```
> adset <- fb_ad_create_adset(  
+   name = 'My budget for promoting my BURN lightning talk',  
+   campaign_group_id = campaign,  
+   bid_type = 'CPC',  
+   bid_info = list(CLICKS = 42),  
+   campaign_status = 'ACTIVE',  
+   lifetime_budget = 4200,  
+   end_time = as.numeric(as.POSIXct('2015-09-30')),  
+   targeting = target)
```

We need a creative!

Get an image for the ad:

```
> img <- 'R-tricolor.png'  
> download.file('http://photos3.meetupstatic.com/photos/event/b/c/2/c/global_399768172.jpeg', img)
```

Upload to Facebook:

```
> img <- fb_ad_create_image(img = img)
```

Take a note on the returned hash:

```
> str(img)  
List of 3  
 $ filename: chr "R-tricolor.png"  
 $ hash      : chr "b8568971f4673927cd9e823645d9f991"  
 $ url       : chr "https://scontent.xx.fbcdn.net/hads-xat1/t45.1600-4/12061211_6040312377320_402554620_n.png"  
> img <- img$hash
```

Define the creative:

```
> creative <- fb_ad_create_creative(  
+   fbacc,  
+   name = 'HUN text and tricolor R logo as the ad creatives',  
+   body = paste(  
+     'Ingyenes előadás a Facebook hirdetések',  
+     'optimalizálásáról a következő R meetupon'),  
+   title = 'Online marketing R felhasználóknak',  
+   object_url = 'http://www.meetup.com/Budapest-Users-of-R-Network/events/225331913/',  
+   image_hash = img$hash)
```

Create the ad

```
> ad <- fb_ad_create_ad(  
+   name           = 'An ad -- right from the R console',  
+   campaign_id    = adset,  
+   creative       = creative)
```



CARD.com

Sponsored · 🌐

Ingyenes előadás a Facebook hirdetések optimalizálásáról a következő R meetupon



Online marketing R felhasználóknak

WWW.MEETUP.COM

Sign Up

Like Comment ➦ Share

A/B testing

```
> images <- c('http://www.r-project.org/Rlogo.png',
+             'http://photos3.meetupstatic.com/photos/theme_head/4/e/c/c/full_3200172.jpeg')

> for (l in 1:length(images)) {
+
+   ## download image
+   img <- tempfile(fileext = '.png'); download.file(images[l], img)
+
+   ## resize, then upload image
+   system(paste('convert', img, '-resize 350x150^', img))
+   img <- fbad_create_image(fbacc, img)
+
+   ## create creative
+   url <- 'http://www.meetup.com/Budapest-Users-of-R-Network/events/225331913/'
+   creative <- fbad_create_creative(
+     fbacc,
+     name = 'Online marketing R felhasználóknak',
+     body = paste(
+       'Ingyenes előadás a Facebook hirdetések',
+       'optimalizálásáról a következő R meetupon'),
+     title = 'Online marketing R felhasználóknak',
+     object_url = url,
+     image_hash = img$hash)
+
+   ## create ad
+   ad <- fbad_create_ad(
+     fbacc,
+     name = paste0('An ad -- right from the R console (', toupper(letters[l]), ')'),
+     campaign_id = adset,
+     creative = creative)
+ }
```

Performance metrics

```
> res <- fbRads::fbad_request(path = paste0(fbacc$sacct_path, 'adgroupstats'),
+                             params = list(access_token = fbacc$access_token))

> jsonlite::fromJSON(res)$data[, c('impressions', 'unique_impressions',
+                                   'clicks', 'unique_clicks', 'spent')]
  impressions unique_impressions clicks unique_clicks spent
1    127330         43389         64          60    1457
2    124150         42862         70          69    1395
3    118887         41497         48          46    1351

> power.prop.test(p1 = 48 / 118887, p2 = 70 / 124150, power = 0.5, sig.level = 0.05)
```

Two-sample comparison of proportions power calculation

```
      n = 144959.8
    p1 = 0.0004037447
    p2 = 0.0005638341
sig.level = 0.05
  power = 0.5
alternative = two.sided
```

```
> fisher.test(data.frame(B = c(46, 118887), A = c(69, 124150)), conf.int = FALSE)
```

Fisher's Exact Test for Count Data

```
data: data.frame(B = c(46, 118887), A = c(69, 124150))
p-value = 0.06187
alternative hypothesis: true odds ratio is not equal to 1
sample estimates:
odds ratio
0.696183
```

<https://github.com/cardcorp/fbRads>

