HELLO FRESH

EveryPlate

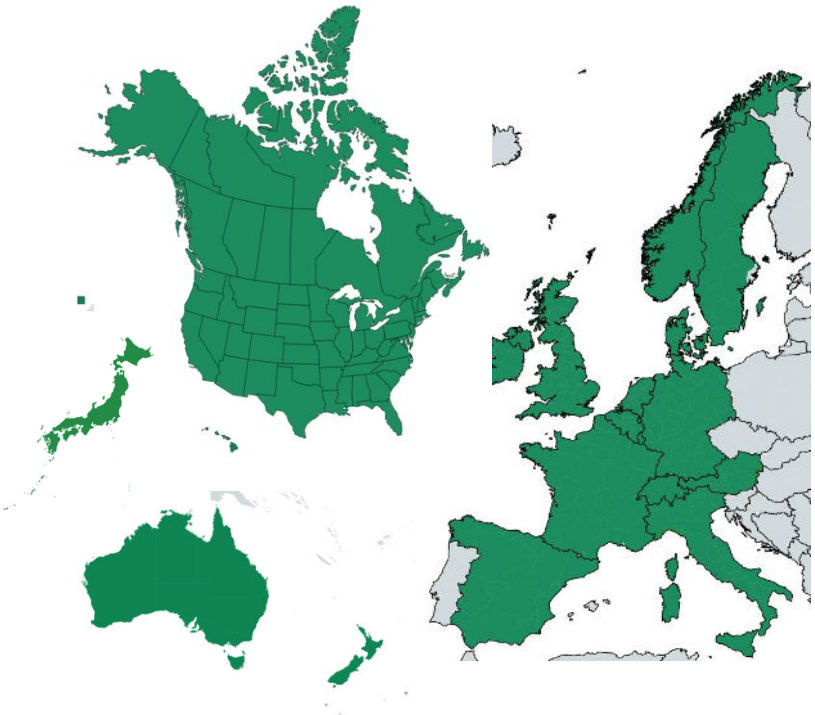
GREEN CHEF

chefs plate

FACTOR_

youfoodz

How our implementation of GTM server, helped us achieve more control over our ~~tracking~~ measurement



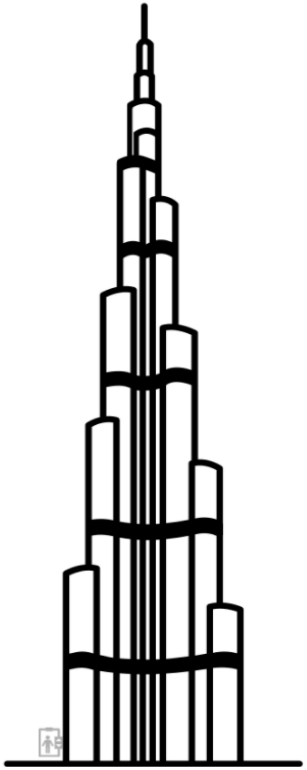
UMA DIGITAL ANALYTICS

HELLO FRESH

GTM server, the elevator pitch (Burj Khalifa edition)

Why should we consider server side tagging?

- Environment that we own and control
- Improvements on client/browser side performance
- Potential improvements to User privacy and security
- Data enrichment/manipulation
- Data is collected in a first party basis



HELLO FRESH

What was our purpose as the “data collection people”?

HELLO FRESH

What did we want to do with our events?

One container to rule them all,

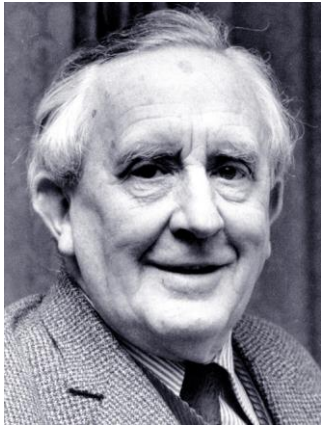
one dataLayer.push() to find them,

one tag to bring them all

and in the Google Cloud bind them

That sounds cool, I should write a short story about it

J.R.R. Tolkien



HELLO FRESH

Server Side tracking implementation in Hellofresh brands

What did we need?

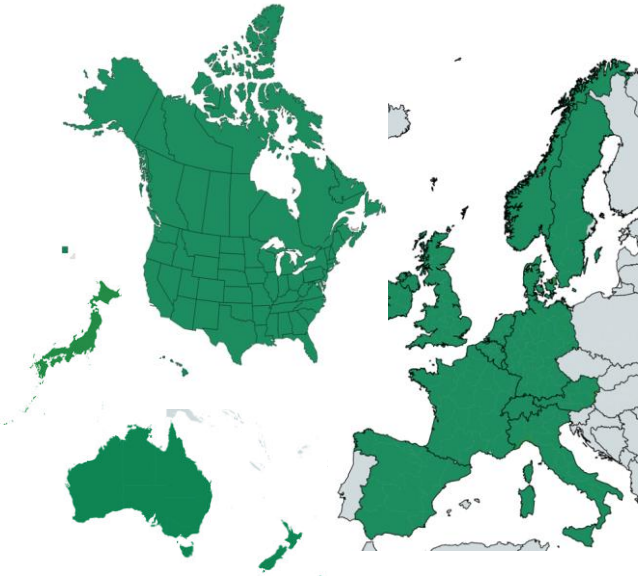
1. A cloud **infrastructure capable of handling +1000 hits per second** (We record over 3 Billion hits/events per month)

2. Servers **physically located either inside or near each of the 17 countries we serve** (as of October 2022), for both **low latency, GDPR compliance purposes and reducing network egress costs.**

3. A way to do **all of our tagging implementation (UA, GA4, Mkt vendors) on GTM server in a single server container**, in order to standardize/centralize them.

4. Start implementing **marketing vendors tracking on the server side** (Affiliates, Facebook, Tiktok, Google Ads, etc)

5. **Serve both GTM.js and GTAG.js scripts from a custom URL** on our servers



HELLO FRESH EveryPlate GREEN CHEF chefs plate FACTOR_ youfoodz

HELLO FRESH

What did we do for each of our needs?

A cloud infrastructure capable of handling +1000 hits per second

HELLO FRESH

What did we do for each of our needs?

1. A cloud server infrastructure capable of handling +1000 hits per second (We record over 3 Billion events per month)

- Initially we used **Google App engine** in Google Cloud Platform
  - **Easy to configure**, it is the default configuration of GTM server
  - It is **auto scalable** (up to whatever nº of instances we set up)
  - It is **relatively cheap** (about 40$/month per instance)

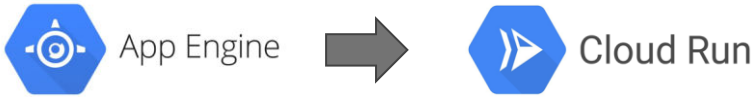
- We had about **100 App Engine instances running** at any given time

- We had setup the auto scaling rules to go up to 4 times the baseline number, to be able to handle traffic peaks with ease.

- We had **alerts telling us if that autoscaling is about to reach its limits**. So we can act accordingly.

What did we do for each of our needs?

We migrated the infrastructure to Cloud Run after 14 months. Why?



1. Only **one App Engine configuration per GCP project**.
   That meant having to manage 20+ different projects (one per website/app)

2. App engine is **not available in all GCP regions**.

3. App engine **instance creation times are too high** (2+ minutes) for sudden peaks of traffic

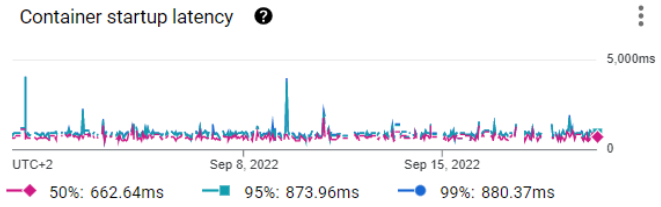
4. App Engine can also **run into resource exhaustion issues depending on region**, where even the minimum number of instances set up cannot be met by the platform

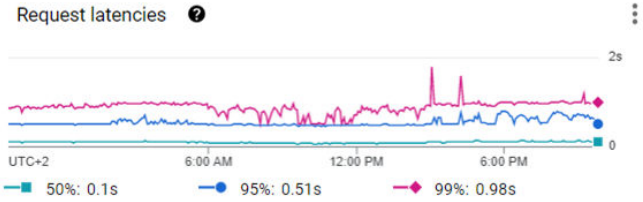
5. **No easy way to load balance the traffic** between different geo-zones

What did we do for each of our needs?

The aftermath

1. A **single GCP project** with shared resources, policies, service accounts, etc.
2. **19 regions serving traffic** VS 8 in App engine
3. Instances in Cloud Run take **less than a second to be up and running.**
4. **No resources issues in over 2 months.**
5. **Improved Network latency times**, less than 140ms in Avg VS >220ms in App engine.
6. All regions can serve traffic for any country and **the load balancer takes care of finding the nearest server to the user.**





HELLO FRESH

What did we do for each of our needs?

The aftermath

Only one downside -> COSTS!

- Our GCP bill for GTM server increased about 15% because of this change.
- For a platform that doesn't have continuous traffic the cost can actually reduce using Cloud Run, since it can be scaled down to zero.

What did we do for each of our needs?

Servers physically located either inside or near each of the 18 countries we serve

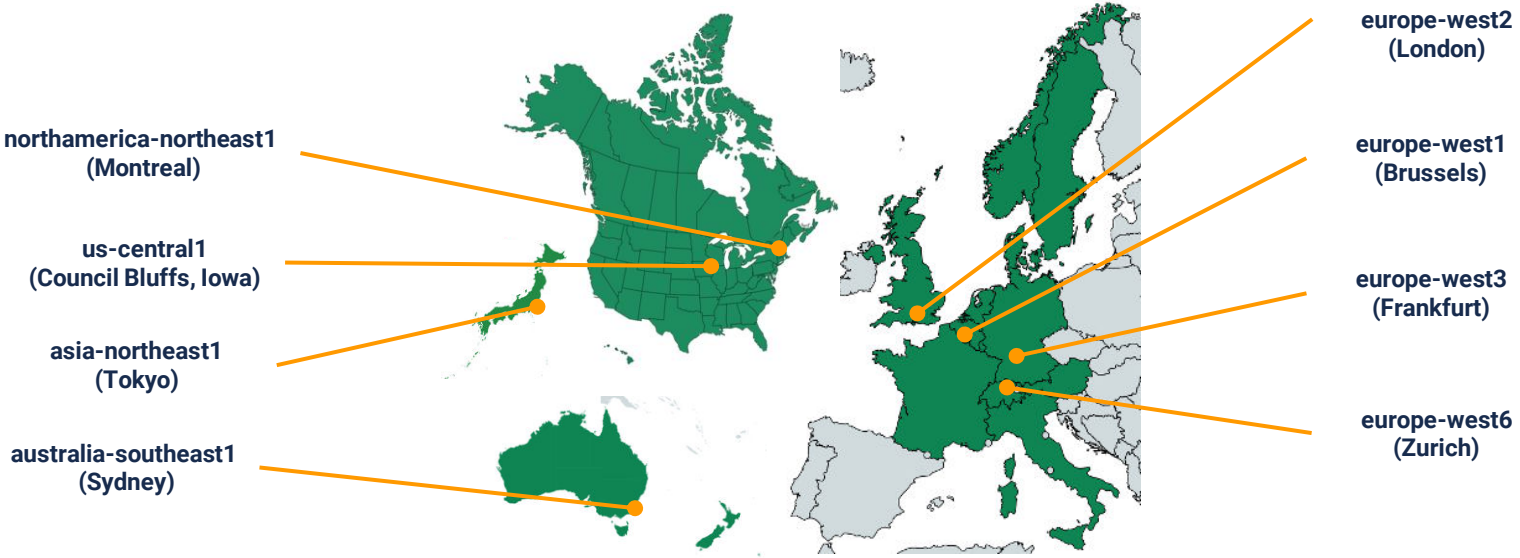
HELLO FRESH

What did we do for each of our needs?

Google Cloud Platform

2. Servers physically located either inside or near each of the 18 countries we serve

- Established **a GCP project for each combination of brand and market** that we provide service
- **Not all GCP regions have App engine available**, we used the closest one to the country in some cases
- We were **using 8 GCP regions and 28 GCP projects** (each assigned to one of the 8 regions)



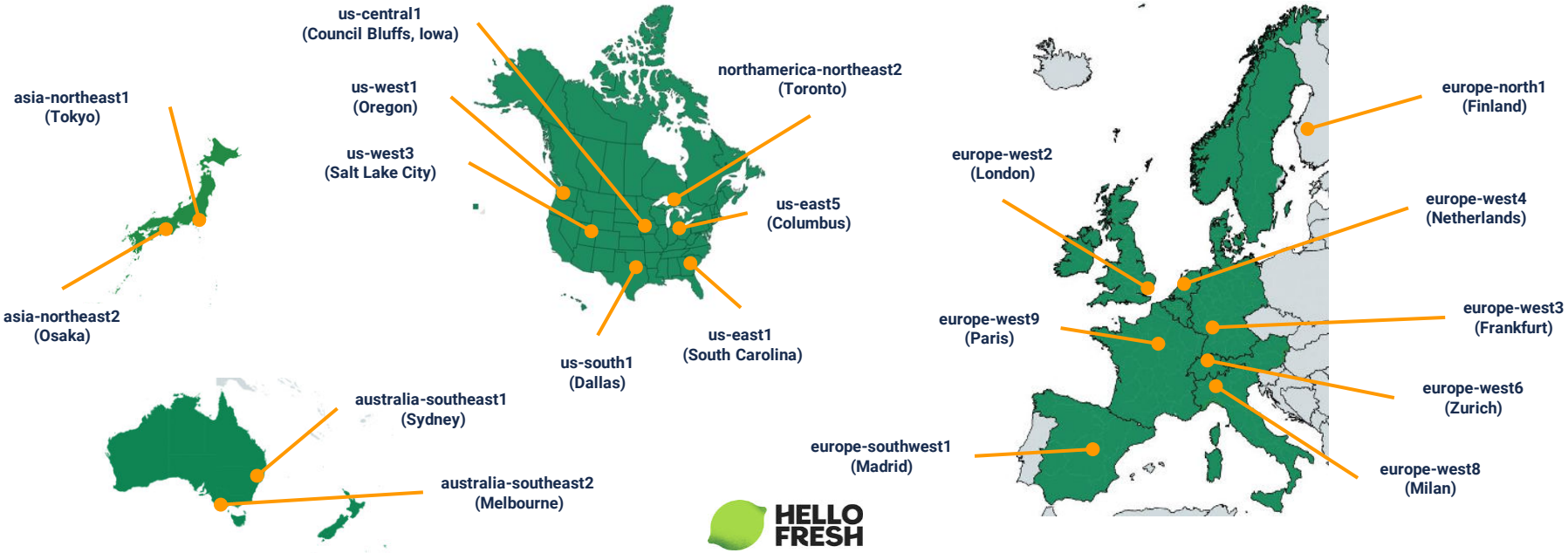
HELLO FRESH

What did we do for each of our needs?

Google Cloud Platform

After migrating to Google Cloud Run a year later

- All resources are now under one GCP project instead of several
- **We have multiple choices of GCP regions within the same country**
- Resources are now load balanced between regions depending on user proximity
- All of them serve the same GTM container



HELLO FRESH

What did we do for each of our needs?

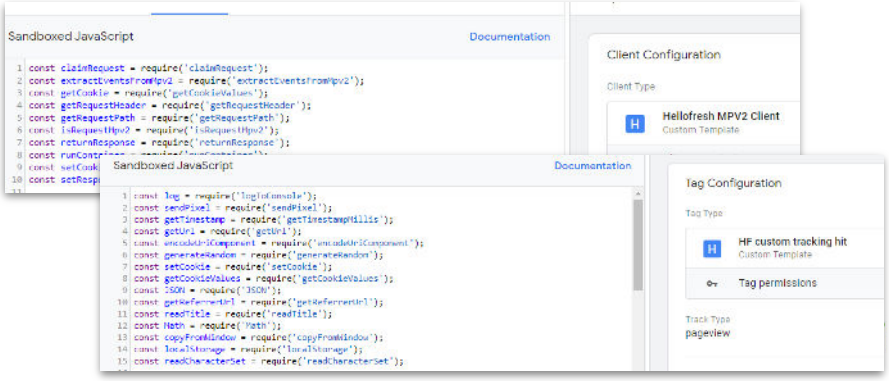
All of our tagging implementation on GTM server in the same server container

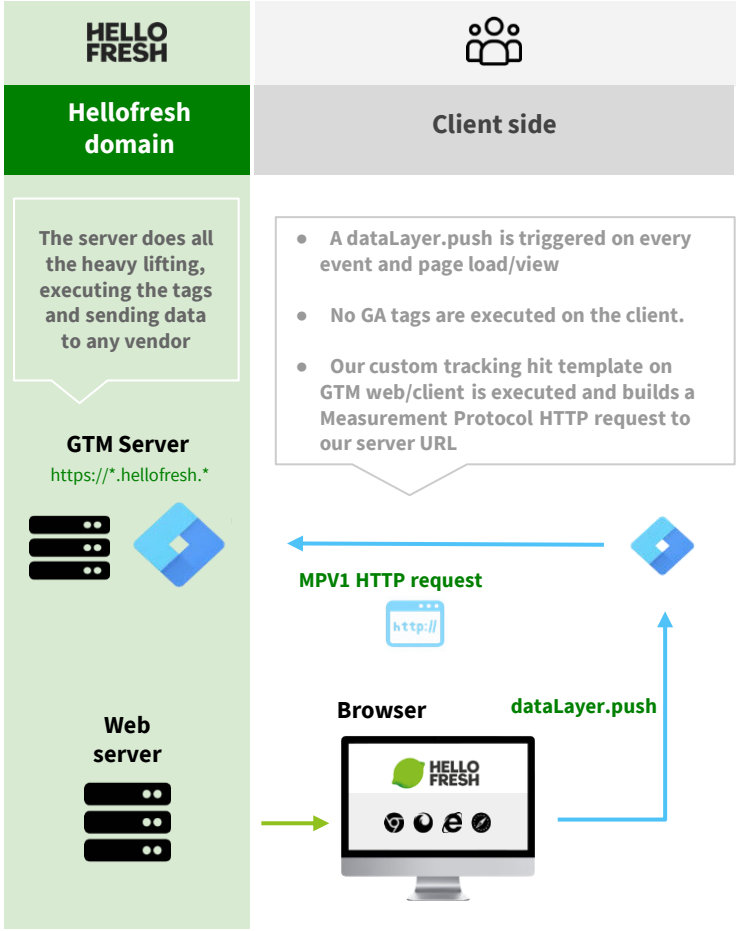
HELLO FRESH

What did we do for each of our needs?

3. All of our tagging implementation on GTM server in the same server container, in order to standardize them

- We **built our own data templates in GTM client side**, in order to build the HTTP requests to GTM server, using measurement protocol V1 format.
- In GTM server we get these standard requests with custom built clients and then trigger the tags for UA, GA4, etc

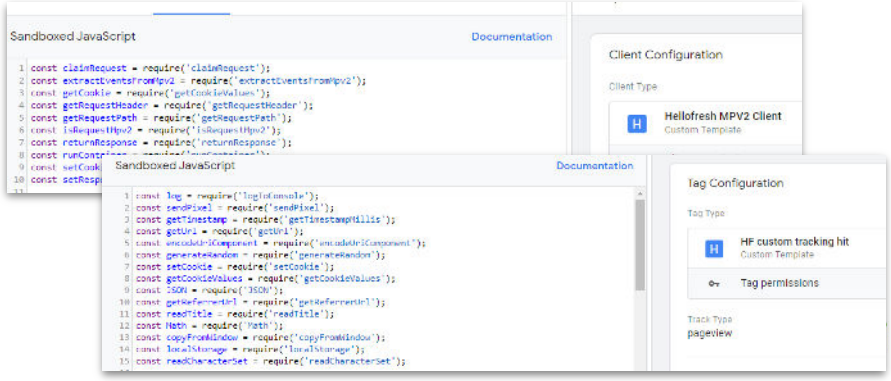


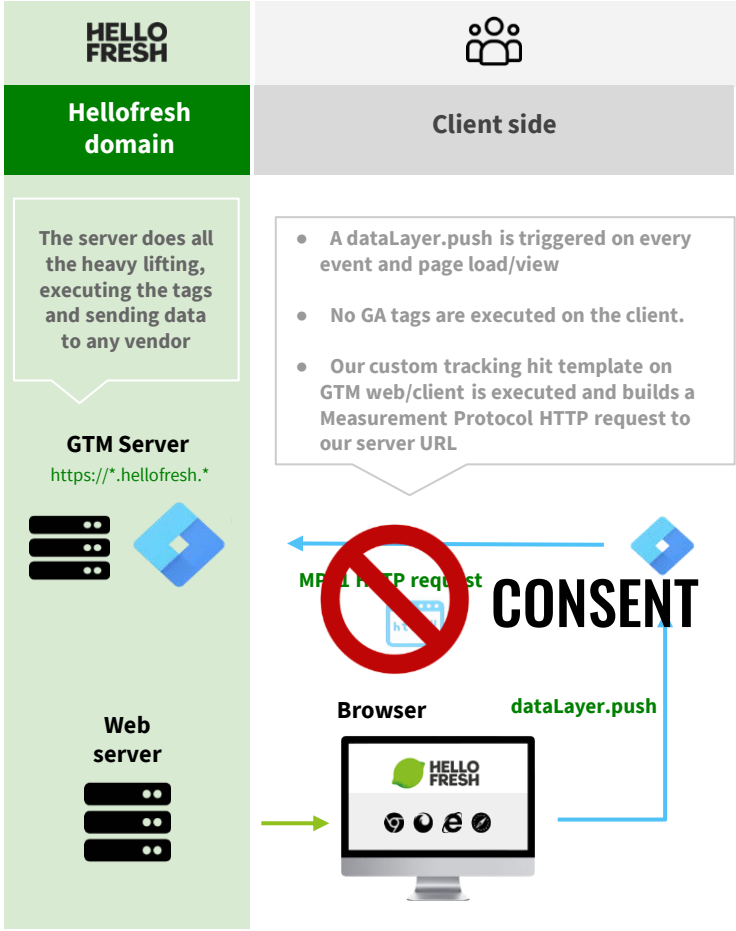


What did we do for each of our needs?

3. All of our tagging implementation on GTM server in the same server container, in order to standardize them

- We **built our own data templates in GTM client side**, in order to build the HTTP requests to GTM server, using measurement protocol V1 format.
- In GTM server we get these standard requests with custom built clients and then trigger the tags for UA, GA4, etc





HELLO FRESH

What did we do for each of our needs?

Because Simo said so

3. All of our tagging implementation on GTM server in the same server container, in order to standardize them

simoahava 9 months ago
If the client doesn't invoke `returnResponse()` then the custom headers are not sent with the default response

simoahava 9 months ago
I guess as the MP client has no reason to return anything it doesn't invoke this

Why didn't you use the built-in Measurement Protocol client in GTM server?

DUMA DIGITAL ANALYTICS

More

- Google Analytics: GA4 — Google Marketing Platform
- Google Analytics: Universal Analytics — Google Marketing Platform
- Google Tag Manager: Web Container — Google Marketing Platform
- Measurement Protocol — Google Marketing Platform
- Measurement Protocol (GA4) — Google Marketing Platform

The built-in client doesn't support sending responses to the browser and we needed the response to be able to set 1st party cookies

simoahava 9 months ago
Ah gotcha - makes sense :)

HELLO FRESH

What did we do for each of our needs?

3. **All of our tagging implementation on GTM server in the same server container, in order to standardize them**

Why didn't you use the transport URL in the native GA tags in GTM browser side?

DIGITAL ANALYTICS

Advanced Configuration

Global Function Name

Transport URL

https://serverurl.hellofresh.com

Use Debug Version

No value set

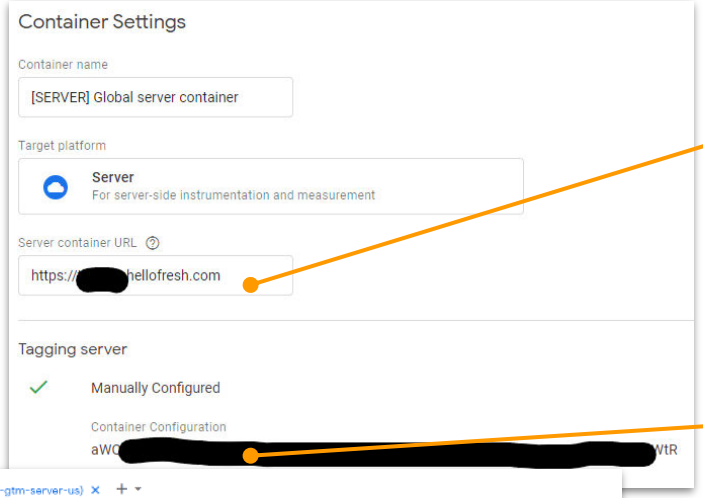
Simple, one of our requirements is not to load the GA scripts at all on the client.

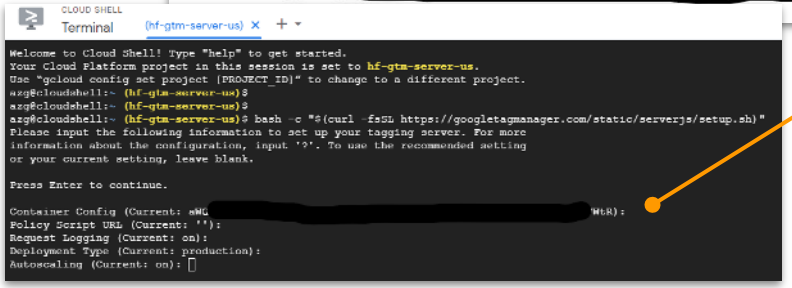
And we liked the challenge :-P

HELLO FRESH

What did we do for each of our needs?

3. All of our tagging implementation on GTM server in the same server container, in order to standardize them





- Whenever we need to test something in a specific site, we just need to change the URL here, to be able to get the right preview.
- All App engine instances for all sites have the exact **same configuration id**.
- This way we have a **single container** controlling all the countries in each brand.
- The main benefit being that **all countries have the exact same tracking setup**

HELLO FRESH

What did we do for each of our needs?

Most marketing vendors tracking on the server side

HELLO FRESH

What did we do for each of our needs?

4. Marketing vendors tracking on the server side (Affiliates, Facebook, Tiktok, Google Ads, etc)

- We have **built our own server tag templates** to communicate with the marketing vendors APIs (Except for Google Ads tags)
- Each of **these templates "piggyback" on the events generated by the custom MPV1 client** on the GTM server and fill the requests with the data needed
- **Specific focus on protecting PII**

Tag Templates

Name ↑
Bigquery data stream
FB Conversion API
ROKT API
Taboola API
TikTok Events API
Verizon

Client IP address	{{Request - IP address - GDPR redacted}}
Email	{{Hashed Email - GDPR redacted}}

Tag Configuration

Tag Type

FB Conversion API
GTM Server

Tag permissions — 7 permissions

Event Name Setup Method
Override

Event Type
Standard
Purchase

API Access Token
{{Lookup - Facebook Access Tokens}}

Facebook Pixel ID
{{Lookup - Facebook ID}}

User Data

Property Name	Property Value
Client user agent	{{Request - User-Agent}}
Client IP address	{{Request - IP address - GDPR redacted}}
Email	{{Hashed Email - GDPR redacted}}
Country	{{Lookup - Shop Country - Lowercase}}

Custom Data

Property Name	Property Value
currency	{{Lookup - Shop Currency}}
contents	{{Facebook - Custom Data - product contents}}
value	{{MPv1 - Ecommerce Revenue}}
country	{{Lookup - Shop Country}}

Firing Triggers

UA MP client trigger - sale
Custom

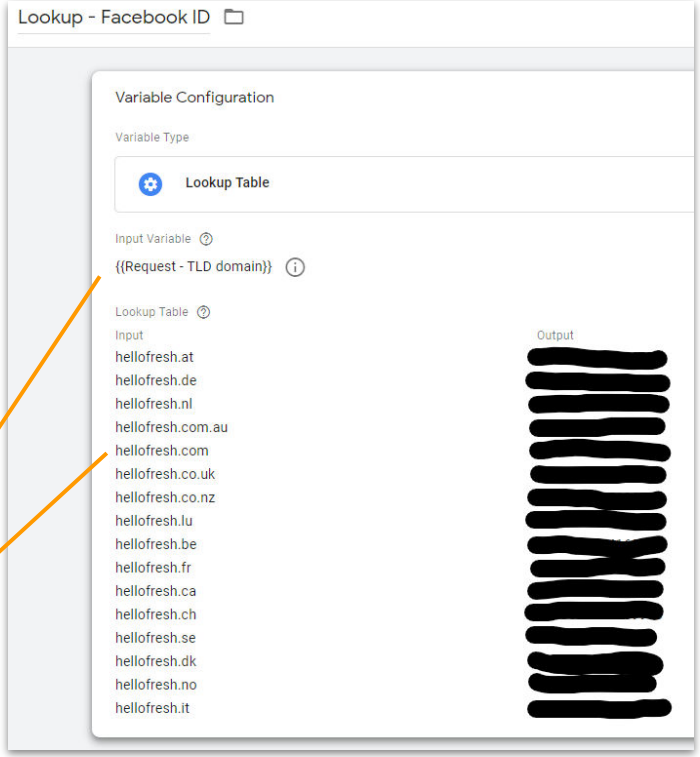
HELLO FRESH

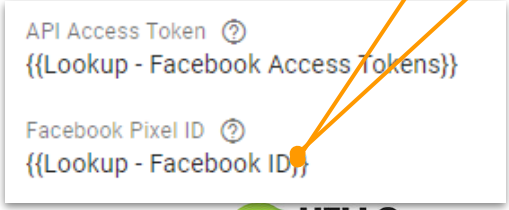
What did we do for each of our needs?

The magic of Lookup tables

4. Start implementing marketing vendors tracking on the server side (Affiliates, Facebook, Tiktok, Google Ads, etc)

- **Template is built in a generic way**, that it can adapt to each market needs, but all using the same base code.
- We have built **Lookup tables for every single variable that is country specific** (Currency, API ids, access tokens, etc)
- This **reduces the implementation time** of a new market/country/site to simply input these ids in the respective lookups and publish the container





HELLO FRESH

What did we do for each of our needs?

Serve both GTM.js and GTAG.js scripts from a custom URL on our servers

HELLO FRESH

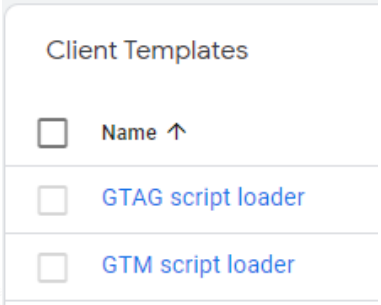
What did we do for each of our needs?

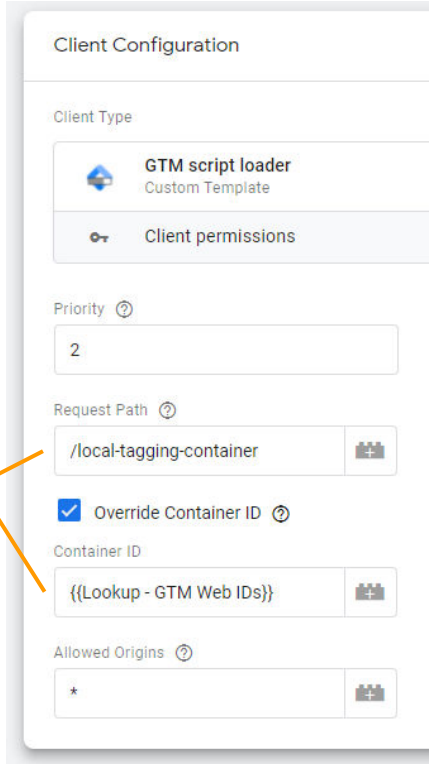
5. Serve GTM.js and GTAG.js scripts from a custom URL on our servers

- Based on the excellent custom GTM loader from Simo Ahava, we have built our own custom GTM server clients, to load both the GTM and GTAG scripts, the latter needed for GA4 specific functionalities on the client/browser.

- We have a standard loading method in all sites now and we don't depend on the developers putting the right container ID on the code, since we determine that directly on the GTM server now.

Now more important than ever, since apparently we cannot even load Google Fonts without triggering EU data agency





HEL FRESH

What did we do for each of our needs?

Some numbers (Just for Hellofresh as a brand)

- From 16 GTM containers to 2 global ones (One on the client and one on the server)
- From 140+ Facebook browser tags (9 events per each of the country containers) to only 9 server tags (one per event we want to capture).
- From 64 Tiktok browser tags, to just 4 server tags
- So far we have removed more than 600 tags from the browser

HELLO FRESH

Server Side tracking implementation in Hellofresh brands

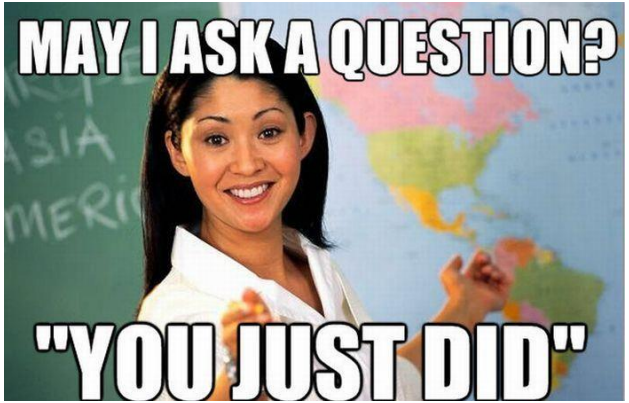
What were the benefits of this?

- **Improvements on our page loading times**, since we started removing different pixels from the browser. Though we still have a long way to go on this topic.
- **Drastic reduction of implementation times** for different tracking vendors across countries. Apart from the initial development of the server templates.
- **Noticeable improvement on the nº of conversions and activities tracked** in both GA and other 3rd party vendors. In the worst of cases we saw an uplift of 6% in the data tracked by GTM server VS parallel implementations on the browser.
- **GDPR compliance and protection of user privacy is *more* under our control** than ever before.

Downsides?

- **Costs of the cloud infrastructure**. At about 40$ per App Engine instance per month. Costs can pile up quite easily.
- **Added maintenance time of the template tags** and adapting to API changes of each vendor
- **Added monitoring time of server resources**, since the infrastructure is in our hands now
- **Single point of failure**, if we introduce a bug in one of our templates. All the global tracking for that vendor might be affected

HELLO FRESH


MAY I ASK A QUESTION?
"YOU JUST DID"